A LINE PHOTOGRAMMETRY ALGORITHM FOR 3D RECTILINEAR OBJECT RECONSTRUCTION

Submitted to the University of Cape Town in fulfilment of the requirements for the Degree of Master of Science in Engineering.

By

Justin Hill

Department of Geomatics

February 1998

DECLARATION

I hereby declare that this thesis is my original work and has not been submitted in any form to another university.

Justin Hill

Abstract

This thesis introduces an alternative formulation for line photogrammetry. The aim was to develop and test a method of computing the position and orientation of a straight line in space using two or more oriented images of that line. The algorithm presented is intended for object reconstruction and is motivated by the need to reconstruct man-made objects in urban areas, such as buildings and the industrial inspection arena. The method aims to obtain a best-fit line through a "pencil of planes". The reconstructed 3D line is defined by two points as opposed to the conventional representation, which uses a point and a direction vector.

The approach to this problem involves the calculation of a projection plane for each image containing the perspective centre and two transformed line-point observations in the image. A least squares adjustment involves fitting a straight line as near as possible to the projection planes from all images simultaneously. The adjusted line is referred to as a best-fitting line through a "pencil of planes" (POP). In this project, a mathematical model was formulated for the application of this concept.

This algorithm was coded and tested on two cases. A set of scanned aerial images of a residential area with a scale of 1:5000 provided the primary test case. Lines delineating three roofs visible in the aerial images were reconstructed using the POP method and compared with ground truth data. The lines reconstructed using the POP method were compared to those reconstructed using an existing method of line photogrammetry, proposed by Mulawa (1988). The second test was based on a set of close-range images captured using a small-format digital camera. Lines delineating the bars of a metal frame generally used as a precise control field for camera calibration, were reconstructed.

In both test cases, χ^2 tests were applied, and the standard deviations calculated. In the aerial case, standard deviations obtained were generally in the region of about 5cm. The ground resolution of the images was 7.5cm. In the close-range case the ground resolution was approximately 1.3mm, and the standard deviations obtained were generally of the order of 0.7mm. Of the lines computed, 84% of the adjustments passed the χ^2 test. The results obtained confirmed that the POP algorithm is a practicable means of adjusting observations to obtain best-fitting 3D lines using observations made in a set of oriented images.

Acknowledgements

I would like to express my sincere gratitude to my supervisors Dr Scott Mason and Professor Heinz Rüther for their help, guidance and enthusiasm in my work. It was a privilege and an inspiration to work with them.

To my fellow post-graduate students, Simon, Ulrike, Justin, Malcolm, Siddique, Jun, Eric, Dirk, Lani, Henty, and staff members Jenny, Julian and Mike, thanks for the help, discussions, advice, laughs, sympathy, friendship and ongoing support. It was a pleasure working with you all.

I would also like to thank Michael, Sydney, Sue, Mrs Atkinson, Joy and Kari of the Department of Geomatics for their help with the many various tasks. To Peter and Patrick, thanks for the humour and support. To Ali and Jean, thanks for the help and encouragement.

To my family, thanks for the endless encouragement and support throughout my school and university careers.

TABLE OF CONTENTS

LIST OF TABLES

TABLE OF FIGURES

1. Introduction

1.1. Aim of Study

The original aim of this study was to identify an effective and efficient method of determining the position and orientation of a straight line in space by using two or more oriented photographs of that line. The method was to avoid the use of traditional point based photogrammetry and to exploit the linearity of straight-line features to obtain a more practical method.

Various existing methods were studied, and it was decided to investigate an alternative approach to the already established methods that had been considered. It became the aim of this thesis to formulate and test this method, and thus determine its viability and potential for application in 3D rectilinear object reconstruction. An algorithm was therefore to be developed, coded and tested on real data. This was done.

This algorithm was then compared with another method proposed by Mulawa and Mikhail (1988), that is based on the use of the coplanarity condition. In order to perform this investigation, both of these algorithms were coded in C++ and tested using real data.

1.2. Scope and Limitations

This thesis endeavours to describe the mathematical model of the POP approach to line photogrammetry, and to provide an interpretation of the results obtained, as well as various findings gathered in the process.

As stated in the project aim, it was intended to develop an algorithm for the POP method, and to code it (in C++) to enable it to be tested. An important part of the testing is the comparison with a well-known and already established method proposed by Mulawa and Mikhail (1988), for which purpose Mulawa's method also had to be coded.

Line photogrammetry for object reconstruction as employed for this project does not include the use of this concept for relative orientation, space resection, or any other task. It is solely for the reconstruction of straight lines in space using images of known relative and absolute orientation. Furthermore, although the theory of line

photogrammetry extends to circles and other conic sections, only straight lines have been dealt with in this project.

Although much of the work done for this thesis involves digital photogrammetry, it is not the intention to introduce or describe the theory or background of digital photogrammetry or photogrammetry in general. General principals of photogrammetry and digital photogrammetry such as the bundle adjustment, direct linear transformation, the fundamentals of charge coupled devices, camera calibration, and absolute and relative orientation are therefore assumed known. For information on these topics, the reader is referred to Cooper and Robson (1996). Only background of line photogrammetry in particular is given in this thesis.

Furthermore, the input required by the algorithms comes in the form of a number of point observations acquired using edge detection software, or points identified manually. There are many different methods for automatically or semi-automatically extracting edges from a digital image, and it is not the task of this thesis to investigate these models, and testing that was done in this project was limited to manual observations.

Lastly, this research was performed during the course of 1996 through December 1997, giving a time limitation of two years for the study. A large portion of this time was dedicated to the study of line photogrammetry in general, and the development of the POP algorithm. Once a model for this method had been formulated, this and Mulawa's algorithm was coded. Programming is a time consuming and ongoing exercise with additional functions and options always being required (no smart interfaces or frills are included!). Due to the time constraints, the test cases were limited to two situations, an aerial, and a close-range case. Most thorough testing was done on the aerial case, as the modeling of urban environments using aerial photography was central to the original motivation for the project. The testing done was considered sufficient to prove the effectiveness of the POP algorithm as a potential method of computing 3D line parameters using a set of oriented images.

1.3. Thesis Outline

To reiterate what was stated in the previous section, the aim of this study was to investigate the probability of the use of a best-fit pencil of planes by comparing it to another established method. This report aims to give a brief background to the subject

of line photogrammetry, describing the various approaches, and then to introduce the theory behind these two methods. Following this, the methods used in testing the algorithms are described, and finally test results are given and analyzed.

Chapter 2: This section on the *Theory of Line Photogrammetry* begins by describing how 3D lines in space can be defined. Various formulations exist for the description of 3D lines in space, and this theory is fundamental for the development of line photogrammetry algorithms.

Secondly in Chapter 2, different formulations for lines in image space are presented. Line photogrammetry models are dependent on which formulations are adopted to describe straight lines in image and object space.

Section 2.3 deals with data acquisition. In this section, it is described how image line-points are observed. Section 2.4 then closes the chapter by explaining the common theories that can be used to relate a straight-line feature in image and object space. Here the coplanarity and collinearity conditions are described, followed by a note on geometry and critical configurations.

Chapter 3 details the algorithm used by Mulawa and Mikhail (1988), followed by an explanation of the method developed in this thesis that makes use of the concept of a "pencil of planes". The concepts and mathematical models used in the two algorithms are described.

Chapter 4 describes the coding of the two algorithms, and the data used in the testing. This data was in the form of oriented and calibrated digital images in which image line-points were observed. Various tests were performed on the POP algorithm including error analyses and comparison with results obtained using Mulawa's method, and given ground truth data. The methods of testing the algorithms are described in this chapter, and results given. A few comments are then made regarding findings during the tests.

Chapter 5 gives an analysis of the results obtained, and discusses a few of the practical characteristics that were observed during the testing.

Chapter 6 concludes by summarizing what was achieved during the project, and presents some final comments.

1.4. Background

Digital photogrammetry has enabled the development of automatic and semi-automatic processes for 3D environment mapping that would not have been possible with analogue photogrammetry. Target detection and identification, point correspondence, interior and exterior orientation, space intersection and resection can all be done automatically or semi-automatically using digital images. Such developments and the related software tools have encouraged progress in new fields of data processing and representation. An example is the 3D visualization of mapped environments using digital terrain and surface models (DTM's and DSM's). These have been used by Baltsavias et al (1995) together with the derived orthoimages in the 3D modeling of buildings. It is stressed by Baltsavias et al that the acquisition of 3D models of man-made objects is greatly important to many users of geoinformation, including planners, geographers, architects etc. An automated or semi-automated 3D modeling process would hence be in demand. Although it is possible to do such modeling, there still exists no quick and simple method, and this has therefore been the focus of ongoing research.

One way of approaching the problem of 3D reconstruction of man-made objects is to choose appropriate features characterizing these objects, and to model the objects by reconstructing these individual features. One problem is the choice of natural primitives for the task. Typical man-made environments are composed of line features and most commonly straight lines, although circles and other conic sections also occur. If one can utilize line features that characterize the man-made objects in environments such as residential areas and industrial plants, the task of modeling that environment can be greatly simplified.

Line photogrammetry has been developed in view of these situations. This now provides a tool that can be used to determine the parameters of straight lines in space photogrammetrically, though independently of point-based photogrammetry being feature-based rather than point-based. Observations of points on the object feature no longer need to correspond with points from other images, and furthermore the increased number of observations on a line feature increases redundancy hence improving accuracy. This is not only an advantage for the user, but also indicates that line photogrammetry has potential as a useful tool in the development of an

automated procedure for object reconstruction. It is primarily this factor that has motivated the study of line photogrammetry in this work as a contribution to the broader goals of the UrbanModeler project (see Section 1.4.2 below) in which semi-automatic building extraction plays an important role.

Line photogrammetry is not restricted to object reconstruction. The general model includes the determination of camera calibration parameters, space resection and relative orientation. In most such cases, results achieved using line photogrammetry have been as good as, or better than those achieved using classical point-based stereo photogrammetry (see Mikhail, 1993; Schwermann, 1994; Petsa and Patias, 1994, 1994a, and 1995; Tommaselli and Tozzi, 1996). A further advantage is that geometric relations between lines in object space such as perpendicularity and parallelity can be used to identify a set of lines that represent more complex 3D objects (Zielinsky, 1993). Prior knowledge of these geometric relations can also be applied in the form of constraints or additional observations in space resection, camera calibration, or object reconstruction.

Line photogrammetry theory is also not restricted to straight lines. Conic sections have been used in space resection and intersection (See Mulawa and Mikhail, 1988; Deren and Guoqing, 1994; Fraser and Dawson, 1996), though it is only straight lines that are of interest in this study. In addition to the generation of 3D urban models a further application of line photogrammetry is the determination of the precise 3D form of rectilinear industrial objects for the purpose of quality control.

1.4.1. Motivation for Using Line Photogrammetry

For certain photogrammetric tasks, line photogrammetry can be a more economical, and sometimes more precise method than conventional point-based photogrammetry. This section describes some of the characteristics of line photogrammetry that provide motivation for further research into the development of the method.

It is easier to automatically detect and robustly extract a line feature in an image than it is to detect an individual untargeted point. Line features are usually characterised by a pattern of grey-level intensity changes in greyscale images and the pixels associated with these intensity changes can be identified using automatic edge detection techniques. Since there are many pixels associated with a single line feature, there is often much redundancy. This enables the location of the feature to be determined to a

greater accuracy than is possible with a single point. Furthermore, since observations of points on a line feature need not correspond with observations from other images, all observations can be included in the adjustment.

For the 3D reconstruction of man-made objects using point-based photogrammetry, the process of "image matching" is performed to identify corresponding points between images (Gruen and Baltsavias, 1988 describe this matching process in detail). This process is time consuming and prone to errors, and the possibility of using line photogrammetry offers a faster, and simpler solution to the 3D-modeling problem.

The correspondence problem is reduced in line photogrammetry, as individual observations of points on a line need not correspond between images. It is simply required that the observations in each image whether in the form of individual points, or a line equation, correspond with the same object line feature. In fact, provided the same line feature is observed in each (oriented) image and the feature is geometrically consistent (e.g. straight line or circle), the images do not need to overlap. This is illustrated in Figure 1, which shows a line in space observed in three images with known orientation. A further advantage is that occlusions are less of a nuisance.

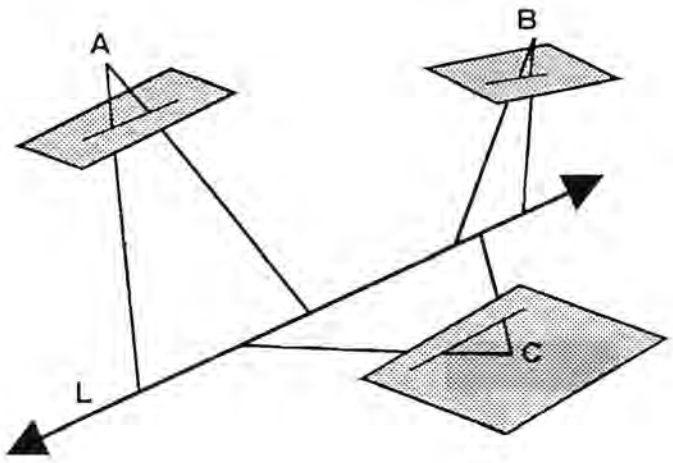


Figure 1: The geometry of observations in Line Photogrammetry

1.4.2. The UrbanModeler Project

Photogrammetry has gone beyond map and orthophoto production, and is used extensively as a data source for geographical information systems, and 3D modeling (see Baltsavias et al, 1995, Mason and Streilein, 1996 and Mason, 1996 for recent work on this topic). Work in this field has been undertaken by the UrbanModeler team at the University of Cape Town. One of the major themes of this project is the development of methods for rapid and low-cost mapping of informal settlements using large scale aerial imagery and digital photogrammetric and image processing techniques. Information gathered in this process can be used in the upgrading and spatial management of informal settlements using geographical information systems. Mason and Rüther (1997), and Mason (1997) detail this concept of an informal settlement GIS (ISGIS).

Research done on 3D modeling of urban environments, within the UrbanModeler project, provided the motivation for this study of line photogrammetry as a potential tool in the process. A common characteristic of dwellings in informal settlements is a rectangular shape with approximately straight edges, and line photogrammetry has potential as a fast and convenient method of reconstructing these dwellings or rooftops, whichever is required.

1.4.3. Previous Work on Line Photogrammetry

Zielinsky (1993) gives a brief chronological account of the development of line photogrammetry. The summary describes the early approaches to the subject, which included work by Masry (1981) and Lugnani (1982), who both used linear features for image orientation. Masry constructed an analytical formulation both for the object feature and the model, and transformed the two features onto each other thereby solving for the exterior orientation parameters. Lugnani (1982) had a similar approach but used curves as well as lines to compute camera orientation parameters by space resection.

In these earlier theories, no direct relation between object features and their 2D image representations existed. In 1988, Mulawa and Mikhail introduced a theory that directly related image and object features by way of the coplanarity condition (described in Section 2.4.1). This provided a direction that was followed in many subsequent studies. Where Mulawa and Mikhail (1988) had recommended a six-

parameter representation, Zielinsky (1993) applied the coplanarity condition approach with a four-parameter straight-line representation. Mikhail (1993) also used a four-parameter representation.

Wilkin (1992) used the coplanarity condition and applied the RANSAC (random sample consensus) procedure combined with a weighted least squares model to make the algorithm more robust when operating on contaminated or noisy data. The result is an algorithm that uses an iterative treatment to detect and remove outliers. This work deals only with the 3D reconstruction of straight lines by image observations from calibrated and oriented cameras.

Schwermann (1994) used a different approach. Lines in object space are represented by two parametric equations that represent the projections of the 3D line onto two suitable planes of the object co-ordinate system (See Section 2.1.2.1). These two relations are substituted into the regular collinearity equations of point photogrammetry to obtain a direct relation between features in image and object space. The use of two parametric equations to describe the 3D line also means that the line is represented by four independent parameters.

Patias, Petsa and Streilein (1995) used the collinearity condition (described in Section 2.4.2) to relate lines in image and object space both for the purpose of relative orientation and space resection. This approach assumes straight-line equations as observables whereas other methods generally use individual point observations (of the line) directly.

Tommasseli and Tozzi (1996) developed a method of performing space resection using linear features with a recursive approach. With a similar model to that of Patias, Petsa and Streilein, this method in addition makes use of Kalman filtering in the space resection problem. The sequentially estimated camera orientation parameters are fed back into the feature extraction process in the image, simplifying a method of feature searching.

More work on the use of line features in photogrammetry exists, though much of it is based on the concepts outlined above. It is evident that various techniques exist for using straight lines in photogrammetry, both for space resection and for the task of computing 3D line equations. Most of these methods make use of a coplanarity condition, or alternatively a form of the collinearity condition.

2. Theory of Line Photogrammetry

This chapter aims to give a background to the theory of line photogrammetry by describing the fundamental concepts. The most important of these is the description of lines in space and their relation to observations on the image plane. Firstly, methods of representing a 3D line in space are described, followed by ways of representing lines in the image and how these may be observed. Finally, the functional models used to relate lines in image and object space by means of the coplanarity condition and collinearity conditions are given.

The theory presented in this chapter is based on already established methods of line photogrammetry and has hence been classified in this thesis as the "Theory of Line Photogrammetry". The method developed in this thesis, based on the concept of a "Pencil of Planes" does not make use of the theory described in this chapter. It therefore made sense to introduce this method separately, and this is done in Chapter 3.2, together with the descriptions of how lines in image and object space are represented. The contents of this chapter are nevertheless important for the reader to gain insight into the subject, and hence to understand the differences between methods.

2.1. Description of Lines in Space

This section introduces three ways of representing a unique line in space. Section 2.1.1 describes a representation that requires six parameters and two constraint equations. Section 2.1.2 describes a representation using a two-plane projection and another using a spherical co-ordinate system, both of which require only four parameters to define a unique line.

2.1.1. Six Parameter Line Definition

2.1.1.1. Point and direction vector

Different formulations are possible for the description of a 3D line, a popular one being a 6-parameter representation as illustrated in Figure 2. This description is used by Mulawa and Mikhail (1988), Wilkin (1992) and Heikkinen (1992). Three of the parameters represent a single point $S(X, Y, Z)$, on the line, and the other three the unsigned direction vector (L, M, N) along the line. In this case, the vector **p** describes an arbitrary point P on the line, and **s** is a vector from the origin to a point S on the

line. The point P is fixed on the line by the arbitrary scalar factor t, and **d** is the direction vector of the line.

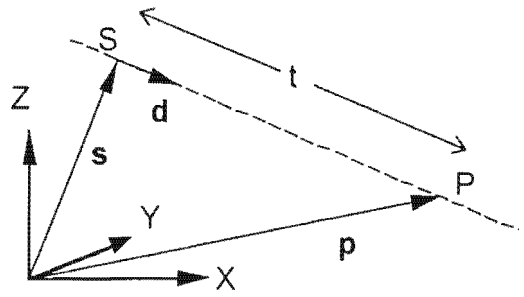


Figure 2: Six-parameter line representation

A common representation of the line is the parametric formulation given in Equation 1.

$$\mathbf{p} = \mathbf{s} + t\mathbf{d}$$

Equation 1

Zielinsky (1993) explains that these six parameters alone are not sufficient to describe a unique 3D line, as a line in space can only be uniquely defined by four degrees of freedom. An explanation for this is that the minimum number of parameters required to represent a 3D line is four (See Section 2.1.2), and hence if six parameters are used, the parameters are not independent. This can be explained geometrically by the fact that any point along the line can be used with a direction vector of any length to give a sufficient representation of a unique 3D line. There is therefore no unique definition for the line. Since there are six parameters describing a point and a direction vector and two of these are dependent on the others, two additional constraints are required for a unique line definition.

Two constraints that can be applied are that the direction vector **d** is assigned unit length (Equation 2), and point S is the point on the line closest to the origin (Equation 3). Here the symbol • indicates a dot or scalar product.

$$|\mathbf{d}| = 1 \leftrightarrow \mathbf{d} \bullet \mathbf{d} = 1^2$$

Equation 2

$$\mathbf{s} \perp \mathbf{d} \leftrightarrow \mathbf{d} \bullet \mathbf{s} = 0$$

Equation 3

2.1.1.2. Two Points in Space

Two points in space $P_1(X, Y, Z)$, and $P_2(X, Y, Z)$ can implicitly represent a straight line. This method of representing a straight line in space is used in the POP method, which is described in detail in Section 3.2.

2.1.2. Four Parameter Line Definition

2.1.2.1. Two-Plane Projection

Schwermann (1994) used a four-parameter representation without the need for a spherical co-ordinate system as used by Zielinsky (see Section 2.1.2.2). The line in space is instead projected onto two suitable planes of the co-ordinate system, creating a pair of 2D lines that can be represented in parametric form. A disadvantage of this approach is that depending on the direction of the line, the gradient could approach infinity in certain projections. Hence a number of cases must be defined and allowed for.

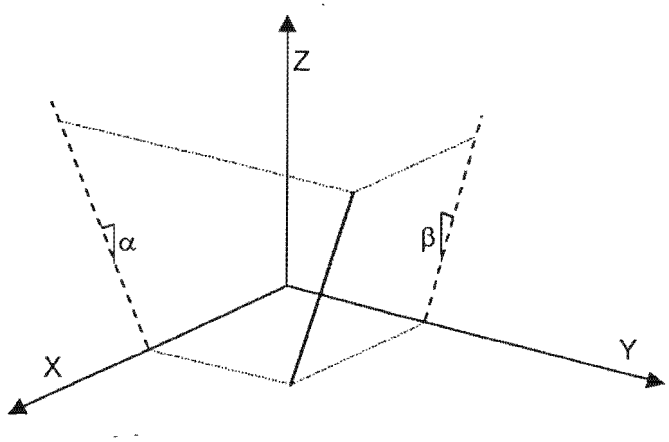


Figure 3: Two-plane (four-parameter) projection of 3D line. Source: Schwermann (1994)

The following equations represent the lines projected onto the X-Y and X-Z planes, the X-Y and Y-Z planes, and the X-Z and Y-Z planes respectively.

$$Y = \alpha_x * X + \gamma_x \quad ; \quad Z = \beta_x * X + \delta_x$$
$$X = \alpha_y * Y + \gamma_y \quad ; \quad Z = \beta_y * Y + \delta_y$$
$$X = \alpha_z * Z + \gamma_z \quad ; \quad Y = \beta_z * Z + \delta_z$$

Equation 4

The pair of equations to be used will depend on the line direction. For example, if the line is nearly parallel to the Z-axis, it is projected onto the X-Z and Y-Z planes.

In order to apply the model for line photogrammetry, the equations above are substituted for X, Y or Z in the conventional collinearity equations (Equation 5) used in point photogrammetry. The resulting equations relate lines in image and object space.

$$x = x_0 - c * \frac{(X - X_0) * r_{11} + (Y - Y_0) * r_{21} + (Z - Z_0) * r_{31}}{(X - X_0) * r_{13} + (Y - Y_0) * r_{23} + (Z - Z_0) * r_{33}}$$
$$y = y_0 - c * \frac{(X - X_0) * r_{12} + (Y - Y_0) * r_{22} + (Z - Z_0) * r_{32}}{(X - X_0) * r_{13} + (Y - Y_0) * r_{23} + (Z - Z_0) * r_{33}}$$

Equation 5

Here, r_{ij} are the elements of the image rotation matrix, X_o, Y_o and Z_o are the object co-ordinates of the perspective centre, x and y are the image co-ordinates of a point in space, and x_o and y_o are the principal point in the image. c is the principal distance.

2.1.2.2. Using a Spherical Co-ordinate System

Some authors, for example Zielinsky (1993), Gülch (1995), and Roberts (in Chang and Aggarwal, 1990) use a spherical co-ordinate system to define a 3D line using four parameters. A point and a direction vector still represent the line, however only four parameters are required for a unique definition i.e. three rotations and a distance from the origin. Figure 4 and the following paragraph illustrate the KTH (Stockholm) approach of representing a 3D line using four parameters as described by Zielinsky (1993).

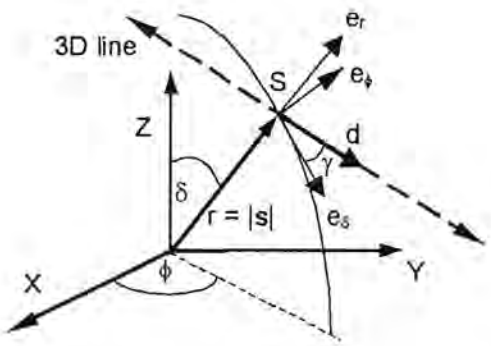


Figure 4: Four-parameter line representation in a spherical co-ordinate system. Source: Zielinsky (1993)

δ Is the angle between the vector **s** and the positive Z-axis and ϕ is the angle between **s** and the positive X-axis. The relations

$$X_S = r \sin\delta \cos\phi$$
$$Y_S = r \sin\delta \sin\phi$$
$$Z_S = r \cos\delta$$

Equation 6

represent the Cartesian co-ordinates of the point S.

To obtain the third rotation angle γ, a spherical co-ordinate system is defined with its origin in the point S, and the three orthonormal base vectors $\mathbf{e}_\delta$, $\mathbf{e}_\phi$, and $\mathbf{e}_r$. Vector $\mathbf{e}_\delta$ is a tangent to the meridian passing through the point S. The vector $\mathbf{e}_\phi$ is the tangent to the parallel circle through point S. $\mathbf{e}_r$ is perpendicular to both $\mathbf{e}_\delta$ and $\mathbf{e}_\phi$, and defined by the vector product ($\mathbf{e}_\delta * \mathbf{e}_\phi$). The angle γ is then defined as the angle between the $\mathbf{e}_\delta$-axis and the direction vector **d** in the $\mathbf{e}_\delta$-$\mathbf{e}_\phi$-plane.

The value r is equivalent to the length of the vector **s**, and represents the distance from the origin to the point S.

Mulawa also described a four-parameter approach (Gülch, 1995), but using two translations and two rotations (y_0, z_0, ϕ, κ). This has not been popular as the method is not well suited to dealing with vertical lines (Gülch, 1995).

2.2. Image Line Definition

In some cases, for example algorithms that make use of the coplanarity condition (described in Section 2.4.1), only single point observations (x, y) are required for each

condition equation in the adjustment. In this case, no specific representation for the line in image space is required. However, the collinearity condition (described in section 2.4.2) as used by Patias et al (1995), and Tomasseli and Tozzi (1996) requires the image line to be represented in one of the following ways.

2.2.1. Two Points

Two points in the form (x_1, y_1), (x_2, y_2) implicitly define the straight line that passes through them.

2.2.2. Slope and Intercept

The parametric form of the line equation is used.

$$y = tx + b \quad \text{or} \quad x = \frac{y}{t} - b'$$

Equation 7

Where t is the slope of the line, and b is the point of intersection with the y-axis.

2.2.3. Two Intercepts

If a and b are the x- and y-intercepts respectively, the equation of a line can be represented in parametric form as follows.

$$\frac{x}{a} + \frac{y}{b} - 1 = 0$$

Equation 8

2.2.4. Trigonometrical

The relation

$$\cos\theta.x + \sin\theta.y - b = 0$$

Equation 9

is used by Tommaseli and Tozzi (1996) and Mikhail (1993), to represent the straight line in the image plane. Here θ represents the angle between the line and the X axis, and b the point of intersection of the line with the Y axis.

2.3. Image Line Observations

Observations in line photogrammetry may either consist of image co-ordinates (x,y) of two or more points on a line in the image, or alternatively the line equation given in one of the forms that appear in Section 2.2, given in the image co-ordinate system. The form of the observations adopted depends on the functional model used. Point observations on a line in an image can be used to calculate parameters in any of the equations given in Section 2.2 if required.

The point observations of a line in an image can either be made manually, or by using an edge detection algorithm. The following two sections describe the process of making image observations in these two ways.

2.3.1. Manual observation

Using image viewing software, the observations (x_i, y_i) can be made simply by recording the x,y pixel co-ordinates of the line-points. The pixel co-ordinate system is based on pixel counts in the horizontal (x) and vertical (y) directions starting at the centre of the upper left hand pixel in the image (See Figure 5).

2.3.2. Observations by Edge Extraction

Various interest operators exist for the detection of grey-level intensity changes in a greyscale image. An automatic or semi-automatic line following routine can be applied to detected edges in order to extract the pixel co-ordinates of points that make up a straight-line feature. For further information on this topic, the reader is referred to Cooper and Robson (1996).

2.3.3. Pixel to Image Transformation

Whether observations are made manually or by automatic edge extraction the observations consist of pixel counts in the horizontal and vertical directions, and thus a transformation from the pixel co-ordinate system to the image co-ordinate system needs to be performed. The pixel co-ordinate system in most image-related software has its origin at the centre of the upper left-hand pixel of the image. The pixel and image co-ordinate systems are illustrated in Figure 5. In the case of a scanned image, the image must first be transformed by rotating until parallel to the pixel rows, and translated so that the position of the upper left pixel is (0,0)

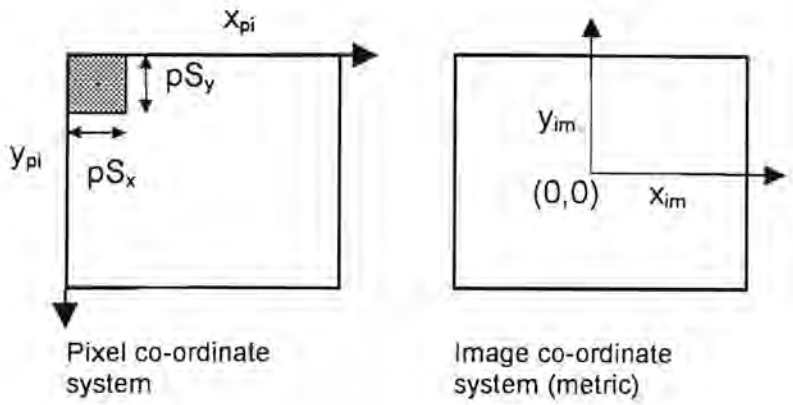


Figure 5: Pixel and image co-ordinate systems

If the centre of the upper left hand pixel is given the pixel co-ordinates of (0, 0), the transformation can be done using the following relations (Smit, 1997):

$$x_{im} = [x_{pix} - \frac{num_x - 1}{2}]pS_x$$
$$y_{im} = [\frac{num_y - 1}{2} - y_{pix}]pS_y$$

Equation 10

in which

x_{im}, y_{im}	are the image co-ordinates
x_{pix}, y_{pix}	are the pixel co-ordinates
num_x	is the total number of pixels in the horizontal direction
num_y	is the total number of pixels in the vertical direction
pS_x, pS_y	are the pixel cell sizes in the x and y directions.

2.3.3.1. Distortion corrections

Observations made directly on the digital image do not only need to be transformed to the image co-ordinate (metric) system, but must also be corrected for distortion parameters. Perturbations in the image resulting from lens distortion, atmospheric refraction, and focal plane distortions are modelled in the process of camera calibration. Fraser (1992) describes four of the principal sources of image distortion in detail. These are symmetric radial distortion, decentering distortion, focal plane

unflatness, and in-plane image distortion. The most significant of these are radial lens distortion, which changes as a function of radial distance from the principal point. Decentering distortion has a less significant effect than radial distortion, but has the important characteristic that it is strongly correlated with the principal point offsets x_p and y_p. This means that decentering distortion can be sufficiently compensated for by a shift in the principal point (x_0, y_0). Focal plane unflatness is distortion in the focal plane perpendicular to the plane of the image, and in-plane image distortion is the perturbation in image co-ordinates within the image plane. A detailed description of the causes of image distortion, as well as techniques in camera calibration is given in Fraser (1992), and Fryer (1996).

Most significant are the principal point offsets, x_p and y_p (see Figure 6). These interior orientation parameters represent the offset of the principal point from the image centre. In some cases, x_p and y_p are the only significant corrections that need to be applied to image observations, however this is dependent on the camera system being used and the precision requirements of the task.

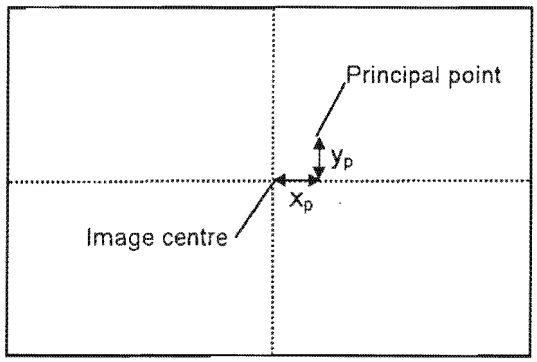


Figure 6: Principal point offsets x_p and y_p

The effect of these parameters is accounted for by applying corrections to the observations using the following equations:

$$x = x_{observed} - x_p + d_x$$
$$y = y_{observed} - y_p + d_y$$

Equation 11

where x and y are the observations to be used in the adjustment, and d_x and d_y are the corrections to be applied for the above mentioned distortions.

2.4. Image-Object Relation

Not all line photogrammetry algorithms directly relate features in image and object space. The coplanarity and collinearity conditions are both used in least squares models that relate the image and object features directly. These two formulations are explained in this chapter.

2.4.1. Coplanarity Condition

The most common functional model for line photogrammetry applies the coplanarity condition to relate line-points in image and object space (e.g. Mulawa and Mikhail, 1988; Wilkin, 1992; Zielinsky, 1993; Heikkinen, 1992). Figure 7 illustrates Mulawa's coplanarity constraint using the notation given in Zielinsky (1993). The vector **d**, the observed direction vector **m**, and the vector formed by the perspective centre O and a point on the line S, are coplanar (Mulawa and Mikhail, 1988).

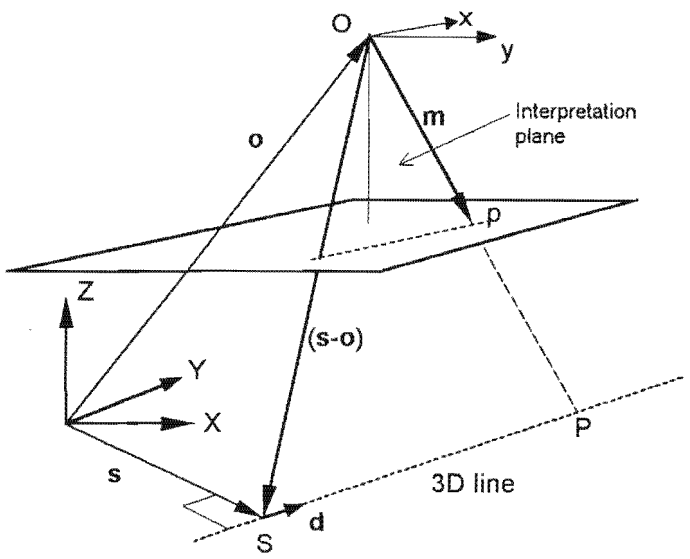


Figure 7: Mulawa's coplanarity constraint

The symbol O is the perspective centre, and **o** is the vector from the origin of the co-ordinate system to the perspective centre. S is the point on the 3D line that is closest to the origin and **s** is the vector from the origin to that point. The vector **m** is the observation ray (formed by O and p) through the point P, an arbitrary point on the object line. **(s-o)** is the vector from the perspective centre O to S, and **d** is the

direction vector along the 3D line. Note the assumption made that the points O, p and P are collinear, i.e.

$$\overrightarrow{\mathrm{Op}} \;//\; \overrightarrow{\mathrm{OP}}$$

Equation 12

The coplanarity condition can be expressed by the relation given in Equation 13.

$$\left|\mathbf{s}-\mathbf{o},\quad \mathbf{d},\quad \mathbf{m}\right| = 0$$

or

$$\begin{vmatrix} X_S - X_O & Y_S - Y_O & Z_S - Z_O \\ L & M & N \\ X_i & Y_i & Z_i \end{vmatrix} = 0$$

Equation 13

Where L, M and N are the components of **d**, and

$$\mathbf{m} = \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \lambda \mathbf{R}^{\mathrm{T}} \begin{bmatrix} x \\ y \\ -c \end{bmatrix}$$

Equation 14

The expanded version of Equation 13 is given in the Appendix. **R** is the image rotation matrix defined by rotations about the Y, X, and Z axes respectively (i.e. $R_{\phi\omega\kappa}$). Here ϕ, ω, and κ represent rotation angles about Y, X, and Z respectively. λ is the scale factor. The values x and y are image co-ordinates corrected for the effects of image distortion, and c is the principal distance, which is held fixed in the adjustment.

Each observed ray (**m**) of a linear feature provides a single condition equation. The unique determination of a straight linear feature in space requires two photogrammetric rays from each of two images. This provides four condition equations of the type shown in Equation 13. These, combined with the two constraints given in Equation 2 and Equation 3, yield six equations. Since six independent

parameters define the line (Mulawa and Mikhail, 1988), a unique solution for these descriptors can be obtained

2.4.2. The Collinearity Condition

The coplanarity condition requires three particular vectors to be coplanar. The vector joining the perspective centre to the line centre, the direction vector of the 3D line, and observation ray for a single arbitrary point on the line. The collinearity condition uses the complete 2D line in image space l rather than a single line-point observation, and hence the need for a suitable representation of this line. The line in image space can be represented by any of the relations given in Section 2.2.

Petsa and Patias (1994), Petsa et al (1995) and Tomasseli and Tozzi (1996) used the collinearity condition as an alternative to the coplanarity condition. The model is based on the equivalence between the vector normal to the interpretation plane (or projection plane) in the image space and the vector normal to the rotated interpretation plane in the object space. An interpretation plane is defined as the plane containing the line in object space (L), the projected line in image space (l), and the perspective centre of the camera (O) (Tomasseli and Tozzi, 1996). This plane would be equivalent to the plane formed by the points O, S and P in Figure 7, however it is represented differently.

In Figure 8 below, the image line l is represented by two intercepts (a and b).

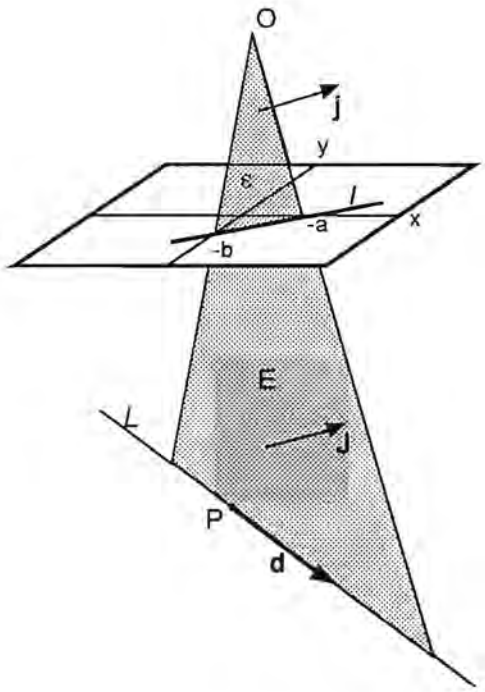


Figure 8: The collinearity condition

The object line L is described by the parametric equation

$$\mathbf{q} = \mathbf{p} + s\mathbf{d}$$

Equation 15

where

$\mathbf{q}^T = [X\text{-}X_0, Y\text{-}Y_0, Z\text{-}Z_0]$ for any point (X, Y, Z) of line L.

$\mathbf{p}^T = [X\text{-}X_p, Y\text{-}Y_p, Z\text{-}Z_p]$ for a reference point P on line L (not related to point S in Figure 7).

$\mathbf{d}^T = [L, M, N]$ the direction vector of line L.

s = a scalar parameter

The direction vector of the 3D line **d**, and the vector from the perspective centre to a reference point P on the line form a plane E. Plane E is represented by the perpendicular vector **J** and is given by

$$\mathbf{q}(\mathbf{p} \times \mathbf{d}) = \mathbf{q}^T\mathbf{J} = 0$$

Equation 16

The image line l and the perspective centre O form a second plane ε, which is represented by the perpendicular **j**. The equation for the vector **j** is dependent on the formulation used to represent the line l. Only one representation will be given here, and the chosen relation will be that which corresponds with the form used in Figure 8 and given in Section 2.2.3

$$\mathbf{j}^T = [-cb, -ca, ab + bx_0 + ay_0]$$

Equation 17

Here (x_0, y_0) is the principal point. The collinearity equation applied to the two vectors **j** and **J** sets the condition that the planes ε and E are parallel by requiring that the two perpendiculars are collinear. Since two parallel planes passing through the same point (the perspective centre O in this case) are coincident, the collinearity condition (Equation 18) is satisfied when the planes ε and E are coincident.

$$\mathbf{j} = \lambda \mathbf{R} \mathbf{J}$$

Equation 18

In the equation above **R** is the image rotation matrix and λ is a scale factor.

2.4.3. Critical Configurations

The accuracy with which a 3D line can be defined depends on the sensor configuration.

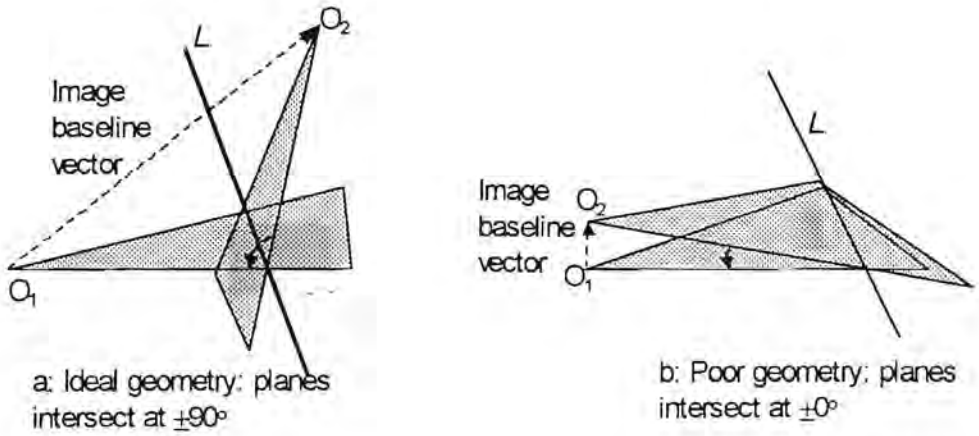


Figure 9: Image geometry for straight-line parameter determination

Best results will be obtained when the 3D line L in object space runs perpendicular to the baseline vector between camera stations (Figure 9a). In this situation, the line position determination is strong in all directions. If the projection planes are nearly parallel (see Figure 9b), the ability to determine the line position in the direction of the image projection planes is weak. If the line joining the perspective centres of all images is coplanar with the line in object space, that is if the baseline vector between images is parallel to the line in space, L cannot be determined.

In this chapter the reader has been introduced to various methods of representing lines in space (3D), lines in an image (2D), and two ways of directly relating the two (coplanarity condition and collinearity condition). In Chapter 3, Mulawa's method is explained in detail, including the least squares adjustment model, which is also implemented in the method of determining a best-fit line through a pencil of planes (POP method). The POP method is also explained in detail in Chapter 3.

3. Description of Algorithms

Two methods of computing the parameters of a 3D line in space by means of a least squares adjustment are presented in this chapter. In Section 3.1, an already established algorithm, referred to here as "Mulawa's method", is described in detail. This is followed in Section 3.2 by a description of the POP method (A method of determining the best-fitting line through a pencil of planes), which was developed during the course of this project.

3.1. Mulawa's Method

3.1.1. Concept

Mulawa's method is based on the coplanarity condition. The underlying concept of this is explained in Section 2.4.1.

The coplanarity condition (Equation 13) is used to relate the points observed on a line in image space to the line in object space. One such observation can be represented in a single condition equation. The advantage of this approach is that the co-ordinates of the point P, are not included in the adjustment (Mulawa and Mikhail, 1988). These would be "nuisance parameters" in the method used by Petsa et al (1995) for example, as the co-ordinates of point P are unknown in the adjustment, yet do not form part of the solution.

The least squares model used is the so called general case with additional constraints, the same model as that used for the best-fitting line through a pencil of planes described in Section 3.2. Mikhail (1976, pp. 214-216) gives a complete derivation of this model.

A single condition equation is simply the expanded coplanarity condition (Equation 13, which is again given here) formulated for each observation.

$$\begin{vmatrix} X_S - X_O & Y_S - Y_O & Z_S - Z_O \\ L & M & N \\ X_i & Y_i & Z_i \end{vmatrix} = 0$$

The condition equation, when linearized, takes the form

$$\mathbf{A}_{c,u}\mathbf{x}_{u,1} + \mathbf{B}_{c,n}\mathbf{v}_{n,1} = \mathbf{f}_{c,1}$$

Equation 19

n = number of image observations where (x_i, y_i) counts as one observation. Hence n=2j where j = number of images.

u = number of unknown line parameters = 6

c = number of condition equations = n = 2j

f = coplanarity values = current value of the condition equation

A = matrix of derivatives of the condition equations with respect to the unknowns

B = matrix of derivatives of the condition equations with respect to the observations

x = solution vector = corrections to parameters

The weight matrices $\mathbf{P}_{ll}$ and $\mathbf{P}_{xx}$ are introduced where $\mathbf{P}_{ll}$ contains the weights of the observations, and $\mathbf{P}_{xx}$ the weights of the unknowns. Equation 20 shows how the elements p_i of the matrix $\mathbf{P}_{ll}$ are calculated, where s_o is the standard deviation of unit weight a priori, and σ_i the standard deviation of the respective observation.

$$p_i = \frac{s_0^2}{\sigma_i^2}$$

Equation 20

If no specific weighting is used, $\mathbf{P}_{ll}$ is an identity matrix, and $\mathbf{P}_{xx}$ a matrix of zeros. As weights for the condition equations, the matrix $\mathbf{W}_e$ is introduced (Wilkin, 1992) as

$$\mathbf{W}_e = \left(\mathbf{B}\mathbf{P}_{ll}^{-1}\mathbf{B}^T\right)^{-1}$$

Equation 21

The matrix $\mathbf{B}\mathbf{P}^{-1}\mathbf{B}^T$ has a diagonal structure and contains "pseudo weights".

In Section 2.1.1, a description was given of the six-parameter line representation using a point and a direction vector. In this section (2.1.1), the need for constraints is explained and the two constraints given (Equation 2 and Equation 3) are the ones that are applied here. Linearized, the constraint equations form the relation

$$\mathbf{C}_{s,u}\mathbf{x}_{u,1} = \mathbf{g}_{s,1}$$

Equation 22

in which **C** is the matrix of partial derivatives of the constraint equations with respect to the unknowns, and s is the number of constraints = 2.

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 2L & 2M & 2N \\ L & M & N & S_x & S_y & S_z \end{bmatrix}$$

Equation 23

and

$$\mathbf{g} = \begin{bmatrix} 1-\left(L^2+M^2+N^2\right) \\ -\left(LS_x+MS_y+NS_z\right) \end{bmatrix}$$

Equation 24

Then, the normal equations are formulated as

$$\begin{bmatrix} \mathbf{A}^{\mathrm{T}}\mathbf{W}_e\mathbf{A}+\mathbf{P}_{xx} & \mathbf{C}^{\mathrm{T}} \\ \mathbf{C} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \mathbf{x} \\ \mathbf{k}_c \end{bmatrix} = \begin{bmatrix} \mathbf{A}^{\mathrm{T}}\mathbf{W}_e\mathbf{f} \\ \mathbf{g} \end{bmatrix}$$

Equation 25

Here, $[\mathbf{x}, \mathbf{k}_c]^{\mathrm{T}}$ is the solution vector, in which $\mathbf{k}_c$ is a vector of correlates that are unimportant in subsequent calculations. The residual vector **v** is calculated by

$$\mathbf{v} = \mathbf{P}_{ll}^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{k}$$

Equation 26

where

$$\mathbf{k} = \mathbf{W}_e\left(-\mathbf{A}\mathbf{x}+\mathbf{f}\right)$$

Equation 27

The a posteriori reference variance is then

$$\sigma_0^2 = \frac{\mathbf{v}^{\mathrm{T}}\mathbf{P}_{ll}\mathbf{v}}{r}$$

Equation 28

where the redundancy of the model r is found using

$$r = c - (u - s)$$

Equation 29

The cofactor matrix $\mathbf{Q}_{xx}$ is given by

$$\mathbf{Q}_{xx} = \begin{bmatrix} \mathbf{A}^T \mathbf{W}_e \mathbf{A} + \mathbf{P}_{xx} & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix}^{-1}$$

Equation 30

The variance-covariance matrix Σ_{xx} is then

$$\Sigma_{xx} = \sigma_0^2 \mathbf{Q}_{xx}$$

Equation 31

and the standard deviations of the unknowns are

$$\sigma_x = \sigma_0 \sqrt{\mathbf{Q}_{xx}}$$

Equation 32

3.1.2. Calculation of provisional values

Provisional values for the line parameters can be computed using two observations from two images. The described method is used by Wilkin(1992). Six provisional values are required, i.e. the co-ordinates of the point S(X, Y, Z), and the components of the direction vector **d** (L, M, N).

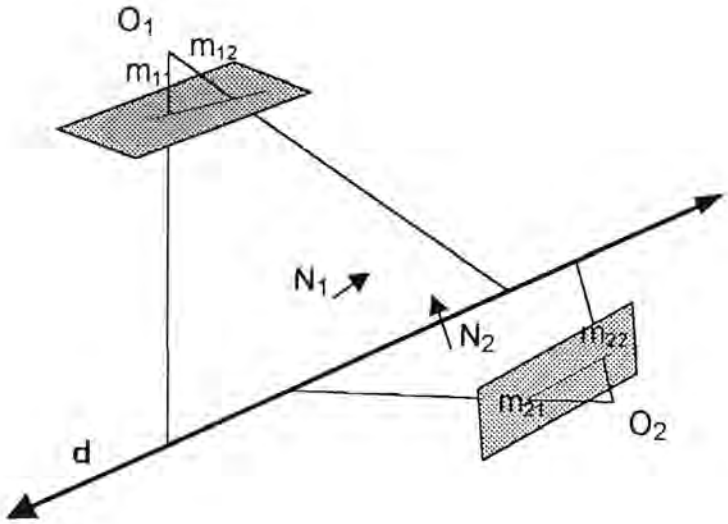


Figure 10: Space geometry in the closed-form solution. Source: Wilkin (1992)

The four observation rays m_{ij} are transformed into object space using

$$m_{ij} = \lambda \mathbf{R}_i \begin{bmatrix} x_{ij} \\ y_{ij} \\ -c_i \end{bmatrix}$$

Equation 33

Projection planes are generated using the two image rays from each image. These planes are represented by their normal vectors, N_i.

$$\mathbf{N}_i = m_{i1} \times m_{i2}$$

Equation 34

The intersection of the projection planes provides the provisional 3D line, and the direction vector **d** of the line is perpendicular to both N_1 and N_2. Thus we obtain the first three provisional values (L, M, N).

$$\mathbf{d} = \frac{N_1 \times N_2}{\|N_1 \times N_2\|}$$

Equation 35

The positions of the projection planes N_1 and N_2 are needed for the calculation of the other three provisional values - the co-ordinates of the line centre point S (X, Y, Z). Using the perspective centres from each image as points that lie on the projection planes, the following equations can be formed.

$$\mathbf{N}_{1x}\mathbf{O}_{1x} + \mathbf{N}_{1y}\mathbf{O}_{1y} + \mathbf{N}_{1z}\mathbf{O}_{1z} - k_1 = 0$$
$$\mathbf{N}_{2x}\mathbf{O}_{2x} + \mathbf{N}_{2y}\mathbf{O}_{2y} + \mathbf{N}_{2z}\mathbf{O}_{2z} - k_2 = 0$$

Equation 36

Since $\mathbf{N}_1$, $\mathbf{N}_2$, $\mathbf{O}_1$, and $\mathbf{O}_2$ have known values, k_1 and k_2 can be calculated. In order to obtain a unique solution, a constraint that the line centre point S is the line-point nearest to the origin is added. This provides the third equation in a linear equation system that can be solved for the three provisional parameters (S_X, S_Y, S_Z).

$$LS_x + MS_y + NS_z = 0$$
$$N_{1x}S_x + N_{1y}S_y + N_{1z}S_z = k_1$$
$$N_{2x}S_x + N_{2y}S_y + N_{2z}S_z = k_2$$

Equation 37

3.2. Best-fitting Line Through a Pencil of Planes

3.2.1. The Concept of a Pencil of Planes

A projection plane was defined earlier as a plane that passes through the perspective centre and two observed points $p_1(x, y)$ and $p_2(x, y)$ on a line in an image. Two non-parallel projection planes from two images separated in space always intersect in a line, however due to the stochastic nature of observations, it is unlikely that projection planes formed by more than two images will intersect in precisely the same 3D line (Figure 11a). If more than two planes intersect each other as illustrated in Figure 11b, the resulting configuration is termed a "pencil of planes".

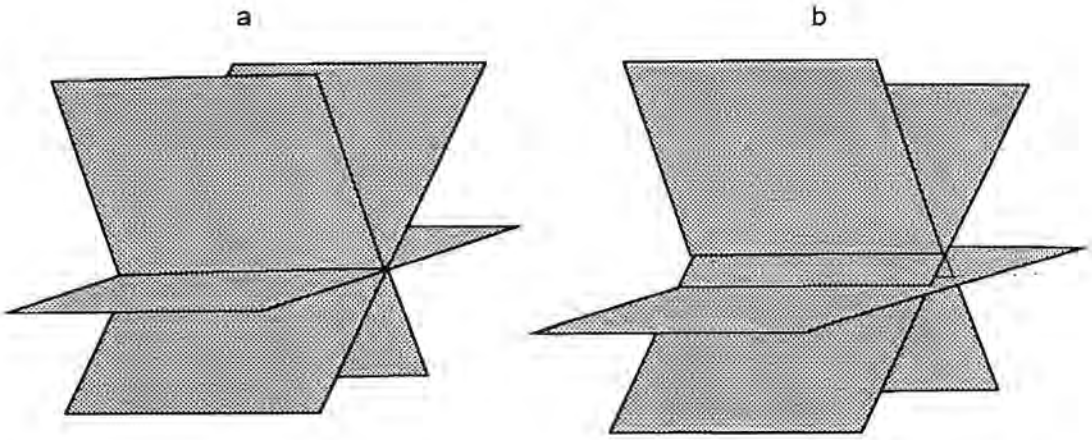


Figure 11: An error-free case (a), and a pencil of planes (b)

If the projection plane intersections do not correspond (Figure 11b), some model is required to determine the most likely position and orientation of the line. The chosen method is to determine that line from which the sum of the squared distances to all projection planes is at a minimum i.e. a "best-fit line through a pencil of planes". This concept is illustrated in Figure 12.

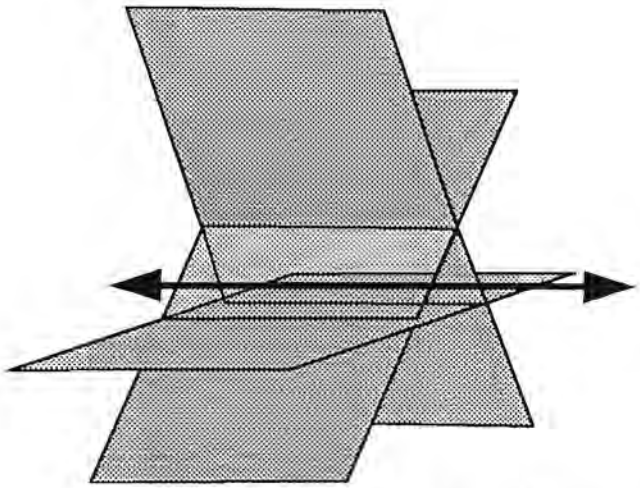


Figure 12: The best-fit 3D line through a pencil of planes

The following two sections explain how lines in image and object space are defined for this particular method, and the least squares adjustment used to determine the final position of a line in space when more than two images of the line are used.

3.2.2. Line Description in Image and Object Space

The line in object space is implicitly represented by two points, (X_1, Y_1, Z_1) and (X_2, Y_2, Z_2) from which any relation defining this line needed in subsequent applications can be derived if required. This is a six-parameter representation, but differs from that described in Section 2.1.1 where a six-parameter representation is described in which a point and a direction vector are used to define the line. In this case, a second point is used in place of a direction vector. The (X, Y, Z) co-ordinates of the points at which the squares of the perpendicular distances to all projection planes are minimised become the parameters that are to represent the line. The positions of both points are adjusted simultaneously. Figure 13 below shows two provisional points with arrows indicating the perpendicular distances to planes.

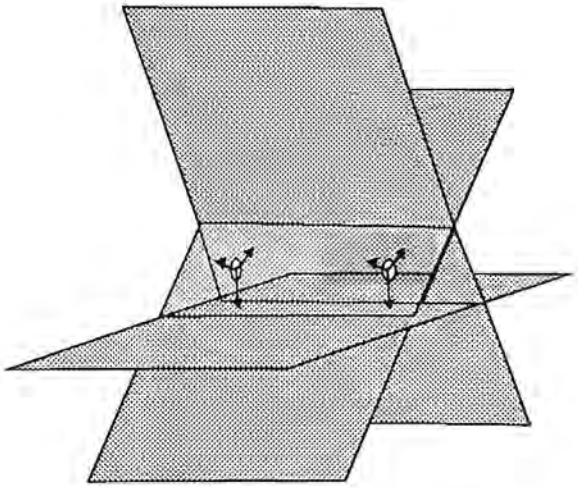


Figure 13: The two provisional line-points

For the reasons outlined in Chapter 2 only four of these six parameters are independent, and two constraints need to be applied in order to obtain a unique solution. Since the two points are treated independently in the adjustment, at least one constraint must apply to each point. Two options exist:

- One co-ordinate of each point is held fixed in the adjustment.
- One co-ordinate of one point is fixed, and a pre-set distance to the second points is imposed.

In both the above cases careful choice of the parameters to be held fixed in the adjustment is needed. The practical reason for holding a parameter fixed is to restrict freedom parallel to the line, as this would result in an infinite number of solutions. The fixed parameter therefore should not correspond with an axis that runs perpendicular to the line. By fixing the value of X, for example, a plane parallel to the Y-Z axes is defined. If the line in question does not run parallel to this plane, a single solution for a point on the line is obtainable. However, if the line runs parallel to the Y, or Z-axes, holding an X-value fixed in the adjustment would result in an infinite number of solutions. Figure 14 illustrates the basic criteria for choosing which parameters to hold fixed.

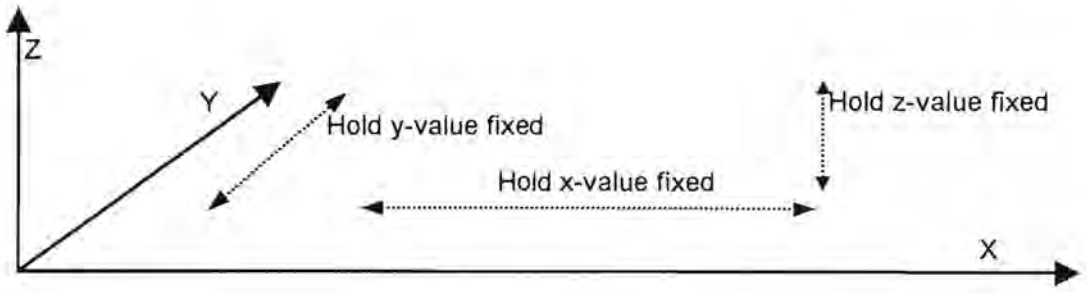


Figure 14: Using line direction to determine which parameters to hold fixed in adjustment

Since the approximate position and orientation of the line are unknown prior to the adjustment, provisional values can used in deciding which parameters are to be held fixed. As a practical rule, the parameter X, Y or Z, which undergoes the greatest change between the two points, should be held fixed.

If a pre-set distance is to be used the actual value chosen is not critical. However, it makes sense for both points to be within the bounds of the area covered by the images where the discrepancies between projection planes will be at a minimum. For this reason, the point positions must roughly correspond with those of the observations in order to obtain meaningful error statistics. A suggested method of calculating provisional values for two line-points is described in Section 3.2.3. Here, provisional line-points corresponding with the observations from one of the images are found. If line endpoints are observed in this case, the provisional, and therefore the final line-points will approximately correspond with the endpoints.

3.2.3. Calculation of Provisional Values for Line-Point Co-ordinates

Provisional values are required for the two points that are to represent the 3D line. The following method was used in computing provisional line-points for the POP algorithm.

Observations from two images are transformed into object space by means of the collinearity equation (Equation 38).

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} + \lambda \mathbf{R}^{\mathrm{T}} \begin{bmatrix} x_i \\ y_i \\ -c \end{bmatrix}$$

Equation 38

In which **R** is the image rotation matrix, and λ is the scale factor. $\mathbf{X}_i$ are the object space co-ordinates of the image observations $\mathbf{p}_i$.

The transformed observation vectors $\mathbf{p}_1$ and $\mathbf{p}_2$ (See Figure 15) in one of these images are then computed using the following equations.

$$\lambda_i = \frac{X_i - X_0}{\sqrt{(X_i - X_0)^2 + (Y_i - Y_0)^2 + (Z_i - Z_0)^2}}$$
$$\mu_i = \frac{Y_i - Y_0}{\sqrt{(X_i - X_0)^2 + (Y_i - Y_0)^2 + (Z_i - Z_0)^2}}$$
$$\nu_i = \frac{Z_i - Y_0}{\sqrt{(X_i - X_0)^2 + (Y_i - Y_0)^2 + (Z_i - Z_0)^2}}$$

Equations 39

The symbols λ, μ and ν represent the X, Y and Z direction cosines of a vector. One projection plane is then calculated using the perspective centre (X_0, Y_0, Z_0), and the two transformed image points (X_i, Y_i, Z_i) from one image. These three points in object space are substituted into a form of the coplanarity condition (Equation 40), which is used to find the equation of the projection plane.

$$\begin{vmatrix} (X - X_0) & (Y - Y_0) & (Z - Z_0) \\ (X_1 - X_0) & (Y_1 - Y_0) & (Z_1 - Z_0) \\ (X_2 - X_0) & (Y_2 - Y_0) & (Z_2 - Z_0) \end{vmatrix} = 0$$

Equation 40

The plane parameters A, B, C are the coefficients of X, Y and Z respectively, and D is the remainder. The expanded version of Equation 40 is given in the Appendix. The projection plane is represented by

$$AX + BY + CZ + D = 0$$

Equation 41

The two line-point observation rays from the second image (Equations 39) are then intersected with the projection plane from the first image (Equation 41), thus providing two points P_1(X, Y, Z) and P_2(X, Y, Z). These intersections are calculated as follows.

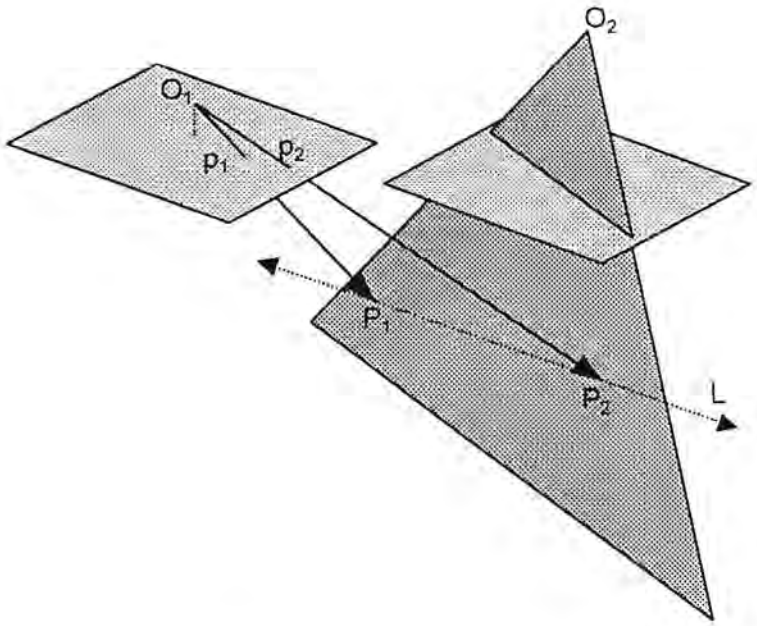


Figure 15: Calculation of provisional line-points

Equation 42 provides the length of the observation ray (OP_i in Figure 15) from the perspective centre of one image (O_1) to the projection plane (through O_2) in the second image. In order to find the length of the vectors O_1P_i illustrated in Figure 15, the following equation is used

$$d_{OP_i} = \frac{-(AX_0 + BY_0 + CZ_0 + D)}{A\lambda_i + B\mu_i + C\nu_i}$$

Equation 42

Then the provisional points P_1 and P_2 can be found by substitution into

$$X_{Pi} = X_0 + d_{OP_i}\lambda_i$$
$$Y_{Pi} = Y_0 + d_{OP_i}\mu_i$$
$$Z_{Pi} = Z_0 + d_{OP_i}\nu_i$$

Equations 43

Once two provisional line-points have been calculated, a least squares adjustment with conditions and constraints can be applied to find the two final points that define a best-fitting line through the pencil of planes. Section 3.2.4 describes the procedure for implementing the adjustment.

3.2.4. Determining the Best-fit Line Through a Pencil of Planes

The two observed image points $p_1(x, y)$ and $p_2(x, y)$ in each image are transformed into the object co-ordinate system using Equation 38. In the case of an edge detection algorithm being used and/or if numerous points on the image line are observed, two arbitrary points on the line derived from a linear regression can substitute as observations.

For each observed point (i = 1, 2) on a line in an image, the collinearity equation (Equation 38, repeated here) is applied.

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} + \lambda \mathbf{R}^{\mathrm{T}} \begin{bmatrix} x_i \\ y_i \\ -c \end{bmatrix}$$

A projection plane is then calculated for every image, by applying the coplanarity condition to the perspective centre (X_0, Y_0, Z_0), and the two transformed image points (X_i, Y_i, Z_i), as described in Section 3.2.3. Again, the plane equation is given in the form of Equation 41.

$$AX + BY + CZ + D = 0$$

Note that the proposed adjustment model does not just require the equation of the plane, but the perpendicular distance from the plane to a point. $d_{Pi\text{-}\{ABCD\}}$ in Equation 44 is the perpendicular distance from point P_i to the plane {A, B, C, D}. X_{Pi}, Y_{Pi} and Z_{Pi} are the provisional values for one of the points chosen to represent the 3D line.

$$d_{P_i\text{-}\{ABCD\}} = \frac{AX_{P_i} + BY_{P_i} + CZ_{P_i} + D}{\pm\sqrt{A^2 + B^2 + C^2}}$$

Equation 44

For every projection plane, one condition equation is formed for each of the points P_1 and P_2 that are to define the object line, i.e.

$$F = \frac{A}{G}X_P + \frac{B}{G}Y_P + \frac{C}{G}Z_P + \frac{D}{G} - d$$

Equation 45

Where

$$G = \pm\sqrt{A^2 + B^2 + C^2}$$

The value *d* in Equation 45 is a fictitious observation of the distance between the provisional point P and the projection plane, and is equated to zero. Although the distances between the points P_1 and P_2 and the projection planes are treated as observed values, they are in fact derived from the actual observed values (x_i, y_i). The value *d* is therefore necessary in the formulation. Equating *d* to zero however, has no effect on the results.

The least squares model used here is the same as that used for Mulawa's algorithm. The model is described in Section 3.1.1 and the matrix formulations are given here. The only differences in applying the model lie in the formulation of the individual matrices, as can be seen from the difference in matrix sizes in Equation 46 and Equation 19.

$$\mathbf{A}_{2j,u}\mathbf{x}_{u,1} + \mathbf{B}_{2j,m}\mathbf{v}_{m,1} = \mathbf{f}_{2j,1}$$

Equation 46

Here j is the number of images. The number of columns in the **B** matrix, and the number of rows in **v** is now equal to *m*, where

$$m = 2n\frac{2j}{2} + 2j = 4j + 2j = 6j$$

Equation 47

Note that the number of image observations is constant, as there are always only two observations (x_1, y_1), and (x_2, y_2) in each image. The value $2n$ in Equation 47 is the total number of parameters in the observations. Hence 2n = 4, since there are two observations with two parameters (x, y) in each image. Furthermore, the number of condition equations is equivalent to twice the number of images *j* in which there are two observations of the line.

The **A** matrix, which contains the partial derivatives of the condition equations F with respect to the unknowns (X_{P1}, Y_{P1}, Z_{P1}, X_{P2}, Y_{P2}, Z_{P2}) is formulated as follows:

$$\mathbf{A} = \begin{bmatrix} \frac{\partial \mathbf{F}_1}{\partial X_{P1}} & \frac{\partial \mathbf{F}_1}{\partial Y_{P1}} & \frac{\partial \mathbf{F}_1}{\partial Z_{P1}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial \mathbf{F}_1}{\partial X_{P2}} & \frac{\partial \mathbf{F}_1}{\partial Y_{P2}} & \frac{\partial \mathbf{F}_1}{\partial Z_{P2}} \\ \frac{\partial \mathbf{F}_2}{\partial X_{P1}} & \frac{\partial \mathbf{F}_2}{\partial Y_{P1}} & \frac{\partial \mathbf{F}_2}{\partial Z_{P1}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial \mathbf{F}_2}{\partial X_{P2}} & \frac{\partial \mathbf{F}_2}{\partial Y_{P2}} & \frac{\partial \mathbf{F}_2}{\partial Z_{P2}} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

Equation 48

The partial derivatives of the condition equations F with respect to the observations are formulated in matrix **B**.

$$\mathbf{B} = \begin{bmatrix} \frac{\partial \mathbf{F}_1}{\partial \mathbf{x}_1} & \frac{\partial \mathbf{F}_1}{\partial y_1} & \frac{\partial \mathbf{F}_1}{\partial \mathbf{x}_2} & \frac{\partial \mathbf{F}_1}{\partial y_2} & 0 & 0 & 0 & 0 & \cdots & -1 & 0 & 0 & 0 & \cdots \\ \frac{\partial \mathbf{F}_1}{\partial \mathbf{x}_1} & \frac{\partial \mathbf{F}_1}{\partial y_1} & \frac{\partial \mathbf{F}_1}{\partial \mathbf{x}_2} & \frac{\partial \mathbf{F}_1}{\partial y_2} & 0 & 0 & 0 & 0 & \cdots & 0 & -1 & 0 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \frac{\partial \mathbf{F}_2}{\partial \mathbf{x}_1} & \frac{\partial \mathbf{F}_2}{\partial y_1} & \frac{\partial \mathbf{F}_2}{\partial \mathbf{x}_2} & \frac{\partial \mathbf{F}_2}{\partial y_2} & \cdots & 0 & 0 & -1 & 0 & \cdots \\ 0 & 0 & 0 & 0 & \frac{\partial \mathbf{F}_2}{\partial \mathbf{x}_1} & \frac{\partial \mathbf{F}_2}{\partial y_1} & \frac{\partial \mathbf{F}_2}{\partial \mathbf{x}_2} & \frac{\partial \mathbf{F}_2}{\partial y_2} & \cdots & 0 & 0 & 0 & -1 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

Equation *49*

In the above equation for **B**, (x_1, y_1, x_2, y_2) are the two image observations in each image.

The vector of unknowns, **x** contains the corrections that apply to the provisional co-ordinates for P_1 and P_2.

$$\mathbf{x}^T = \begin{bmatrix} dX_{P_1} & dY_{P_1} & dZ_{P_1} & dX_{P_2} & dY_{P_2} & dZ_{P_2} \end{bmatrix}$$

Equation 50

The matrix **f** is the vector of discrepancies, and **v** the vector of residuals.

The constraint equations are again given in the form of Equation 22 in which u is the number of unknowns, and s the number of constraints.

$$\mathbf{C}_{s,u}\mathbf{x}_{u,1} = \mathbf{g}_{s,1}$$

Various options exist for the matrix **C**, and the chosen formulation depends on the constraints used. The program used in the test cases imposes a constraint for the distance between the two line-points, P_1 and P_2, and a fixed X, Y, or Z value in one of the points, depending on the orientation of the line. A method of determining which co-ordinate should be held fixed in the adjustment is suggested in Section 3.2.2. The chosen co-ordinate can either be held fixed by weighting, in which case the **C** matrix would have only one row (Equation 53) or alternatively, a second constraint could be used to hold the parameter fixed (Equation 54). The distance between provisional line-points P_1 and P_2 is given in Equation 51.

$$dist^2 = (X_{P2} - X_{P1})^2 + (Y_{P2} - Y_{P1})^2 + (Z_{P2} - Z_{P1})^2$$

Equation 51

The constraint value of the squared distance *DIST* can be given the value of the distance between the two original provisional line-points as calculated in Section 3.2.3, or any suitable value chosen by the user. The discrepancy between the squared value given for DIST, and the squared distance between current line-points, *dist* is then

$$\mathbf{g} = DIST^2 - dist^2$$

Equation 52

The matrix of partial derivatives of the constraint equation with respect to unknowns, **x** is then

$$\mathbf{C} = [-2(X_{P2} - X_{P1}) - 2(Y_{P2} - Y_{P1}) - 2(Z_{P2} - Z_{P1})\, 2(X_{P2} - X_{P1})\, 2(Y_{P2} - Y_{P1})\, 2(Z_{P2} - Z_{P1})]$$

Equation 53

A further option is to hold the parameter X, Y, or Z from one point fixed by adding an additional constraint in the equation **Cx=g** (Equation 22), as opposed to heavily weighting that parameter in the adjustment. For example if the X value of one point is to be held fixed at a particular constraint value, the second constraint equation would be

$$\begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0\end{bmatrix}\mathbf{x} = constraint\ value - current\ value$$

Equation 54

A second constraint equation of this type (Equation 54) holding a parameter of the second point fixed could replace the distance constraint if desired.

It is also possible to hold a parameter X, Y, or Z in both of the two points P_1 and P_2 fixed using a weighted parameter model. In this case there would be no need for constraints. Instead, heavy weights (i.e. large values) would be applied to the elements of the $\mathbf{P}_{xx}$ matrix corresponding to the parameters to be held fixed. However, holding co-ordinates fixed by means of additional constraints is generally a more stable option.

3.2.5. Summary of Steps in Implementing POP Algorithm

The following is a brief summary of procedures and equations involved in implementing the POP algorithm.

- Transform image observations of two points on the line, (x_1,y_1) and (x_2,y_2), in each image into the object space co-ordinate system using Equation 38.

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} + \lambda \mathbf{R}^{\mathrm{T}} \begin{bmatrix} x_i \\ y_i \\ -c \end{bmatrix}$$

- Apply the coplanarity condition (Equation 40) to the object space co-ordinates of the perspective centre (X_0, Y_0, Z_0), and the two observations (X_1, Y_1, Z_1), and (X_2, Y_2, Z_2).

$$\begin{vmatrix} (X - X_0) & (Y - Y_0) & (Z - Z_0) \\ (X_1 - X_0) & (Y_1 - Y_0) & (Z_1 - Z_0) \\ (X_2 - X_0) & (Y_2 - Y_0) & (Z_2 - Z_0) \end{vmatrix} = 0$$

The parameters of the projection plane are the coefficients A, B, C of X, Y and Z respectively in the above equation. The parameter D is the remainder.

$$AX + BY + CZ + D = 0$$

- Obtain two provisional points in space to represent the 3D line:

° Calculate the intersection of the two observation rays from one of the images with the projection plane from a second chosen image (Figure 15). The direction cosines and length of these observation vectors are required for this.

° Determine the three direction cosines of the observation rays using Equations 39.

$$\lambda_i = \frac{X_i - X_0}{\sqrt{(X_i - X_0)^2 + (Y_i - Y_0)^2 + (Z_i - Z_0)^2}}$$
$$\mu_i = \frac{Y_i - Y_0}{\sqrt{(X_i - X_0)^2 + (Y_i - Y_0)^2 + (Z_i - Z_0)^2}}$$
$$\nu_i = \frac{Z_i - Y_0}{\sqrt{(X_i - X_0)^2 + (Y_i - Y_0)^2 + (Z_i - Z_0)^2}}$$

° The length of these vectors is then given by Equation 42.

$$d_{OP_i} = \frac{-(AX_0 + BY_0 + CZ_0 + D)}{A\lambda_i + B\mu_i + C\nu_i}$$

° The two provisional line-points P_1 and P_2 can then be found by means of Equations 43.

$$X_{Pi} = X_0 + d_{OP_i}\lambda_i$$
$$Y_{Pi} = Y_0 + d_{OP_i}\mu_i$$
$$Z_{Pi} = Z_0 + d_{OP_i}\nu_i$$

Once the co-ordinates of the provisional line-points P_1 and P_2 have been determined, the co-ordinate (X, Y or Z) that is to be held fixed in the least squares adjustment can be chosen by means of determining which co-ordinate changes the most between P_1 and P_2. In this summary, it is assumed that the value of X in P_1 is to be held fixed in the adjustment and the distance constraint *DIST* is the calculated distance between the two provisional points P_1 and P_2 determined above.

The next step is to formulate the normal equations.

- Formulate the matrices A (Equation 48) and B (Equation 49), which are made up of the partial derivatives of the condition equations (Equation 45) with respect to the unknowns (X_{P1}, Y_{P1}, Z_{P1}, X_{P2}, Y_{P2}, Z_{P2}), and observations (x_i, y_i) respectively. Also required is the vector of discrepancies, **f**, the vector of residuals, **v**.

° Formulate the additional constraints in the form of Equation 22.

$$\mathbf{C}_{s,u}\mathbf{x}_{u,1} = \mathbf{g}_{s,1}$$

If it is desired to hold the X value of P_1 fixed by weighting, then C and g both have a single row, where

$$\mathbf{C} = [-2(X_{P2} - X_{P1}) - 2(Y_{P2} - Y_{P1}) - 2(Z_{P2} - Z_{P1})\, 2(X_{P2} - X_{P1})\, 2(Y_{P2} - Y_{P1})\, 2(Z_{P2} - Z_{P1})]$$

(Equation 53), and

$$\mathbf{g} = DIST^2 - dist^2$$

(Equation 52).

° The weight matrices W_e (Equation 21) and P_{xx} must be formulated. In this case where the value of X in P_1 is to be held fixed by weighting, the matrix P_{xx} is a matrix of zero's with the exception of the diagonal element that corresponds with the X co-ordinate of P_1 (i.e. the first element). This is given a high value (e.g. 10 000).

In the case where X is to be held fixed by an additional constraint, the equation **Cx=g** acquires a second row of the form given in Equation 54.

$$[1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]\mathbf{x} = constraint\ value - current\ value$$

° Formulate the normal equations in the form of Equation 25, and solve for $[\mathbf{x}\ \mathbf{k}_c]^T$.

$$\begin{bmatrix} \mathbf{A}^T\mathbf{W}_e\mathbf{A} + \mathbf{P}_{xx} & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{k}_c \end{bmatrix} = \begin{bmatrix} \mathbf{A}^T\mathbf{W}_e\mathbf{f} \\ \mathbf{g} \end{bmatrix}$$

• Compute the a posteriori variance using Equation 28.

$$\sigma_0^2 = \frac{\mathbf{v}^T\mathbf{P}_{ll}\mathbf{v}}{r}$$

4. Testing of Algorithm

The POP algorithm was tested using two image sets. One of these was a set of four aerial images of a residential area, and the other is a set of five images taken of a welded iron frame photographed at close range. This chapter describes these two tests in detail.

The aerial images test the performance of the algorithm when used to reconstruct or model building roofs using large-scale aerial photography. In the aerial image-set, the lines that delineate the roofs are very short in relation to the size of the image (typically less than 5% of the image width). The close-range image set was used to test the performance of the algorithm on longer lines and in this case, the line lengths were above 60% of the image width. This test would be of significance in industrial and other close-range applications.

Section 4.1 describes the first step in the testing which involved the coding of the algorithm, and this is followed by Section 4.2 which describes the test data used for the aerial and close-range cases. The testing procedure is described and a discussion of results given in Section 4.3.

4.1. Coding of Algorithms

In order to test the pencil of planes (POP) method of line photogrammetry, the algorithm needed to be coded, and for this, the C++ programming language was used. Mulawa's algorithm was also coded in order to provide an example with which to compare results of the POP method.

As input, both programs require exterior orientation parameters and principal distance, as well as two line-point observations (x_1, y_1, x_2, y_2) of a single line, for each image. The additional two inner orientation parameters, the principal point offsets (x_p, y_p), are applied to the original observations during the transformation from pixel to image co-ordinates. The camera was focussed on infinity for all images, ensuring constant principal distance and image distortion characteristics for all images. In the close-range case, the lens was taped to prevent any changes during image capture.

The POP program accepts only two observations (x,y) from each image, while the program using Mulawa's method can include many observations of a single line from each image.

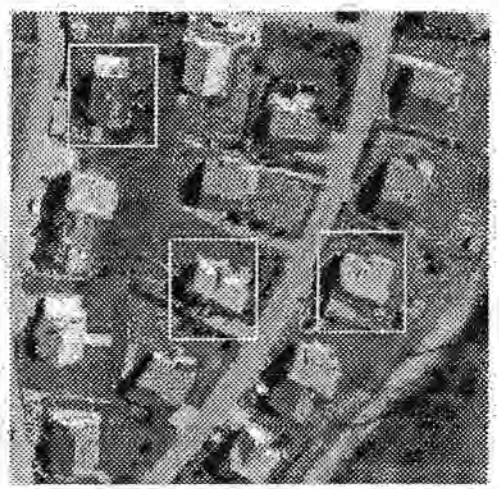
In addition to the adjusted line parameters, the POP program provides an a-posteriori reference variance, computed using Equation 28, and standard deviations of the unknowns found using Equation 32.

4.2. Description of Test Data

4.2.1. Avenches Data Set

Data used for the aerial test case came from a set of photographs taken over Avenches (Switzerland) in 1991 for a project undertaken at ETH, Zürich. The original 23cm by 23cm diapositives had been scanned at 15μm. Only sections of these images portraying the region of interest are provided, and these segments depict part of a residential suburb viewed from four camera stations. The image scale is 1:5000, which gives a ground resolution of approximately .075m.

Interior and exterior orientation parameters for all the camera stations were known from a bundle adjustment. The four images used in this case study are shown below, numbered (from top left) 88, 89, 97, 98 (these numbers were original photograph numbers and are used as references in the Appendix). Frames in these images indicate the specific roofs that were reconstructed.



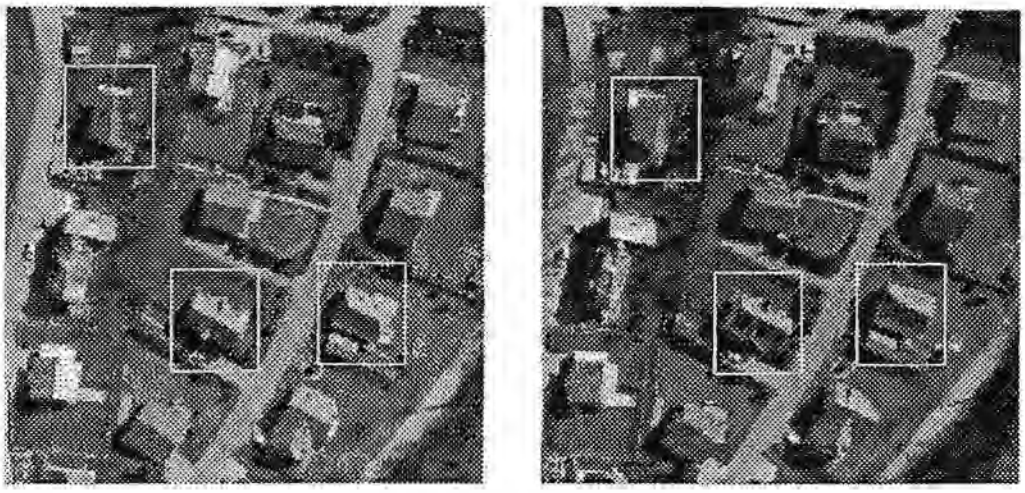


Figure 16: Images used in the testing

4.2.1.1. Acquisition of image-line observations

Manual observations were used for both the aerial and close-range test cases. Individual points could therefore not be identified to greater accuracy than a single pixel.

4.2.2. The Close-Range Data Set

Positions of bars in a welded iron frame generally used for camera calibration were measured in this close-range test. Figure 17 shows the control frame that was used, with the lines that were reconstructed highlighted. An array of precisely co-ordinated circular retro-reflective targets attached to the frame provided control for the images.

Images were captured using a DCS420 digital camera. This camera produces an image 1524×1012 pixels in size, with a square pixel size of approximately 9μm. A ground resolution of 1.3mm was obtained in the images. A distinguishing characteristic of this test was that the photography was not as restricted with respect to choice of camera positions as is the case with aerial photography.

Provisional exterior and interior orientation parameters were computed using a DLT (direct linear transformation). Refined orientation parameters were then computed by means of a bundle adjustment.

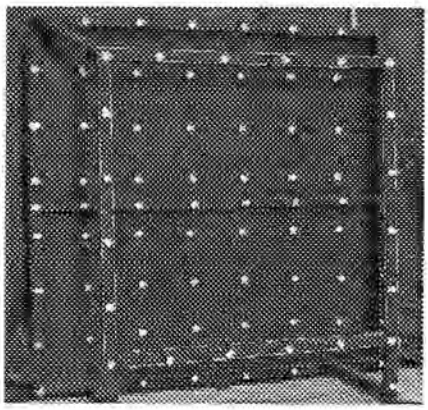


Figure 17: Metal control frame

Five images of the frame were used to reconstruct the eight lines highlighted in Figure 17. Ground truth equations for the reconstructed lines were not available.

4.3. Testing Procedure and Description of Results

Various lines from both the aerial and close-range image sets were reconstructed using the POP method. In the aerial case, the roof outlines of three houses were reconstructed line by line. These roofs are framed in Figure 16. In the close-range case, the edges of four bars on the metal control frame (Figure 17) were reconstructed. The POP algorithm was tested in four ways:

- For each line, an a posteriori reference variance (using Equation 28), and the standard deviations of the X, Y and Z co-ordinates (using Equation 32) of the adjusted points representing the line were calculated. These values are indications of the accuracy expected.
- The adjustment was tested statistically by means of a χ^2 test (See Section 4.3.2), which is an overall test of the functional and stochastic models..
- Mulawa's method was used to compute lines that were also reconstructed using the POP method. The lines obtained by these two methods were directly compared.
- In the aerial case, ground truth data was available, and the lines reconstructed by the POP method were compared with this ground survey data.

The latter two tests indicated above involved the comparison of straight lines in space. Section 4.3.3 describes how the lines were compared.

4.3.1. Standard Deviations of Line-points

The standard deviations referred to are those relating to the X, Y and Z co-ordinates of the two calculated points that represent the 3D line. These standard deviations resulting from adjustments using the POP method are estimates of the accuracy obtained in each adjustment.

Due to the method used to determine provisional points in the POP method, the two line-points calculated in the adjustment approximately correspond with the oriented observation rays from a particular image (see Section 3.2.4). Assuming the observations in that image more or less span the region of interest (e.g. approximate endpoints of a rooftop), the standard deviations output for the determination of the final line-points P_1 and P_2 are directly relevant. To clarify, if a rooftop is to be measured, the user will be interested in the standard deviations of points determined on that line segment defining the rooftop. This differs from other methods such as Mulawa's method in which points elsewhere on the line are calculated.

4.3.2. Testing of Pencil of Planes Algorithm by χ^2 Test

The χ^2 test in this case is based on the model that the expected value for σ_o a posteriori can be estimated by σ_o a priori. However, it is understood that not only a poor ratio between a priori and a posteriori variances can indicate an error in the model, but also too good an agreement between the two (Rüther, 1995). A two tailed test is therefore carried out, which is passed at a certain probability level and for a particular number of degrees of freedom, if the ratio of a priori to a posteriori variances falls between two boundaries of the χ^2 distribution. This can be formulated as follows:

$$\sigma_o^2 \frac{df}{\chi_2^2} \leq s_o^2 \leq \sigma_o^2 \frac{df}{\chi_1^2}$$

Equation 55

where σ_o and s_o are the standard deviations of unit weight a priori and a posteriori respectively. *df* denotes the number of degrees of freedom in the adjustment. Since no specific weights were applied, the parameter σ_o is given the value of the accuracy of the image observations, σ_i. This is justified by Equation 20 which shows the

calculation of weights p_i. Where observations are all weighted equally, $p_i = 1$ and hence $\sigma_o = \sigma_i$.

Although the pixel size in the aerial images is 15μm, it is believed that the accuracy that can be obtained using manual line observations are better, and a value of 7.5μm or 7.5E-6m was therefore assumed for σ_i, giving σ_o the value of 7.5E-6.

At a 99% confidence level the boundary values obtained from the χ^2 distribution were as follows:

Degrees of freedom	Lower Threshold (χ_1^2)	Upper Threshold (χ_2^2)
2 (3 images)	.01	10.597
4 (4 images)	.207	14.86
6 (5 images)	.676	18.548

Table 1: χ^2 values for 2, 4 and 6 degrees of freedom

Substituting these values into Equation 55, the test condition becomes

$$1.06E-11 \le s_o^2 \le 1.12E-8$$

Equation 56

for 2 degrees of freedom (three images used), and

$$1.51E-11 \le s_o^2 \le 1.09E-9$$

Equation 57

for 4 degrees of freedom (four images used). In the close range case, five images are used for some of the adjustments, and here the following condition applies, where σ_0=0.005

$$6.73E-6 \le s_o^2 \le 4.83E-4$$

Equation 58

Tables in the appendix present error statistics for each individual line adjustment. Included in each table is a column indicating, by means of a 1 or zero, whether the adjustment satisfied the conditions of the χ^2 test at a probability level of 99%. Table 2 shows the number of lines that were reconstructed using the POP algorithm and the number of lines passing the test.

Object	Number of lines Reconstructed	Number of Adjustments Passing Test	%
Roof 1	19	14	73
Roof 2	14	13	93
Roof 3	9	9	100
Metal frame	8	6	75

Table 2: Line adjustments passing χ^2 test

It can be seen from the information given in the above table that most adjustments satisfied the conditions of the χ^2 test. This is an indication that the model is correctly formulated. It is likely that the few adjustments that did not pass the test contained an error in the stochastic model. All image observations were given equal weights in the model, and this could have caused adjustments to fail the test if an image with relatively poor observations was included.

4.3.3. Comparing Lines Reconstructed Using the POP, and Mulawa's Method

4.3.3.1. Method of Comparison

The output of the two algorithms, the POP method and Mulawa's method, differ in that the POP method provides two points in space, (P_1, P_2) and Mulawa's method a point in space and a direction vector (S, **d**). Naturally, in the presence of errors, the line passing through P_1 and P_2 is likely to be offset from the point S. This is illustrated in Figure 18 where this discrepancy is represented by the symbol δ. The further point S is from the region of interest, which in this case is assumed to be the line segment (P_1, P_2) representing the approximate endpoints along a rooftop for example, the greater the magnitude of δ. In some situations therefore, the value δ is a misleading representation of the level of correspondence between two lines.

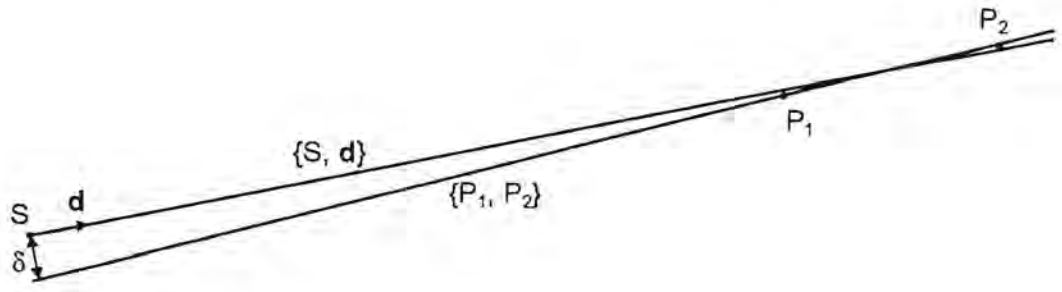


Figure 18: Variable magnitude of discrepancy between lines

Assuming the region of interest to be approximately represented by the line segment (P_1, P_2), a useful comparison between the two lines can be made as follows: firstly by calculating the perpendicular distances from the two points P_1 and P_2 to the second line, and secondly computing the space angle between the two lines $\{P_1, P_2\}$ and $\{S, \mathbf{d}\}$. These measurements are illustrated in Figure 19.

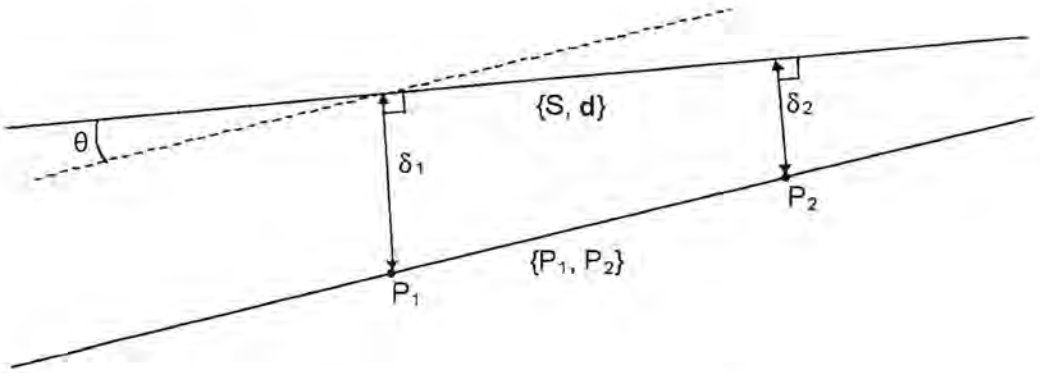


Figure 19: Comparison between calculated lines in space

The perpendicular distance from a point to a line in space can be found using Equation 59.

$$\delta_i = \sqrt{[(X_i - X_c)\mu - (Y_i - Y_c)\lambda]^2 + [(Y_i - Y_c)\nu - (Z_i - Z_c)\mu]^2 + [(Z_i - Z_c)\lambda - (X_i - X_c)\nu]^2}$$

Equation 59

where X_i, Y_i, and Z_i are the co-ordinates of the point P_1 or P_2, and X_S, Y_S, and Z_S the co-ordinates of the line centre point S on the second line. The three direction cosines λ, μ, and ν describe the direction vector $\mathbf{d}$.

In order to find the angle between these two lines, the direction cosines of the line $\{P_1, P_2\}$ must be determined. These are calculated using the same formulae as given in Equations 39 as follows:

$$\lambda = \frac{X_2 - X_1}{\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2}}$$
$$\mu = \frac{Y_2 - Y_1}{\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2}}$$
$$\nu = \frac{Z_2 - Z_1}{\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2}}$$

Equation 60

where X_i, Y_i and Z_i are the co-ordinates of P_1 and P_2. The angle between the two lines can then be found using

$$\cos\theta = \lambda_1\lambda_2 + \mu_1\mu_2 + \nu_1\nu_2$$

Equation 61

where λ_i, μ_i and ν_i for $i = \{1, 2\}$ represent the direction cosines for lines 1 and 2 respectively. It is important that the angle between lines is computed in addition to the two distances. The reason for this is that δ_1 and δ_2 are computed as absolute values due to the square root operation in Equation 59. The effect of this is that one cannot distinguish in what orientation the lines lie in relation to each other. For example, in a situation where the lines cross in space between P_1 and P_2, the lines are closer between the points than may appear given the two distances δ_1 and δ_2. This situation is illustrated in Figure 20.

A further consideration is the length of the measured line segment. If the distance between P_1 and P_2, as determined by the method of computing provisional values in Section 3.2.3, is short in comparison to the original line feature, then the computed line segment must be extrapolated in order to represent the feature. Errors in the computed line segment are therefore exaggerated. This may be the case if obstructions or poor lighting characteristics in a particular image result in only a portion of the line being observable.

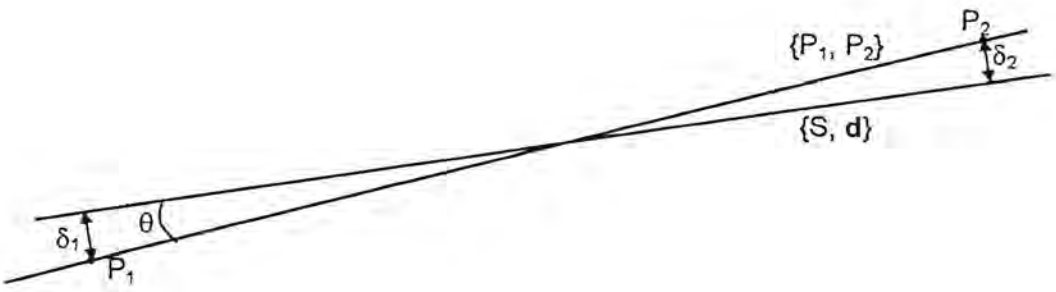


Figure 20: Lines crossing in space

4.3.3.2. Results of the Comparison

A few lines from each test case were used to test the difference in reconstructed lines found using Mulawa's method, and the POP method. The results are tabulated as follows, where lines are labelled according to the figures in Section 4.3.4 :

Test Case	Line	δ_1(m)	δ_2(m)	θ(degrees)
Roof 1	1-2	.006	.003	.12
	2-5	.001	.005	.03
	5-6	.004	.012	.21
Roof 2	2-3	.007	.014	.42
	3-4	.012	.007	.07
	6-13	.002	.003	.06
	6-7	.004	.012	.22
Roof 3	2-7	.007	.003	.05
	1-2	.012	.043	1.1
	5-6	.002	.010	.32
	6-7	.005	.002	.01
Metal Frame	1-2	.02mm	.20mm	.04
	9-10	.11mm	.32mm	.05

Table 3: Comparison between lines reconstructed using Mulawa and POP methods

Considering the image resolution of .075m on the ground in the aerial case, the differences between the lines in all the cases given in Table 3 are notably small. The maximum distance separation given is 0.043m and most differences are sub-centimetre. In the close range case, in which the ground resolution is 1.3mm, the greatest difference is 0.32mm, which is about the same level of accuracy with which the line positions can be determined by physical measurement. This suggests that no significant difference between the results of these two algorithms can be expected.

4.3.4. Comparison with Ground Truth Data

4.3.4.1. Aerial Case

Ground truth data with an estimated accuracy of 2-3 centimetres was given in AutoCad form, and reconstructed lines were superimposed on this data and displayed from various viewpoints to show the level of correspondence between the two. Figure 21, Figure 22 and Figure 23 show the first, second and third roofs from four different viewpoints. Solid lines indicate ground truth data, and the dotted lines have been reconstructed using the POP algorithm. Approximate lengths of the rooftops are given as an indication of scale, and an arrow indicates the North direction in the plan view. Vertices in the plan view are numbered in order to identify each line. A "+" sign indicates lines that were not reconstructed.

The various perspectives given were found to be the most suitable to enable the viewer to visualize the orientation of these lines relative to each other in space. From certain viewpoints, reconstructed lines may fall on the plane through the viewpoint and the ground truth line, and therefore the two lines cannot be differentiated. For this reason, one needs to consider all the viewpoints.

Error statistics, including the results of the χ^2 test for each line are given in the appendix.

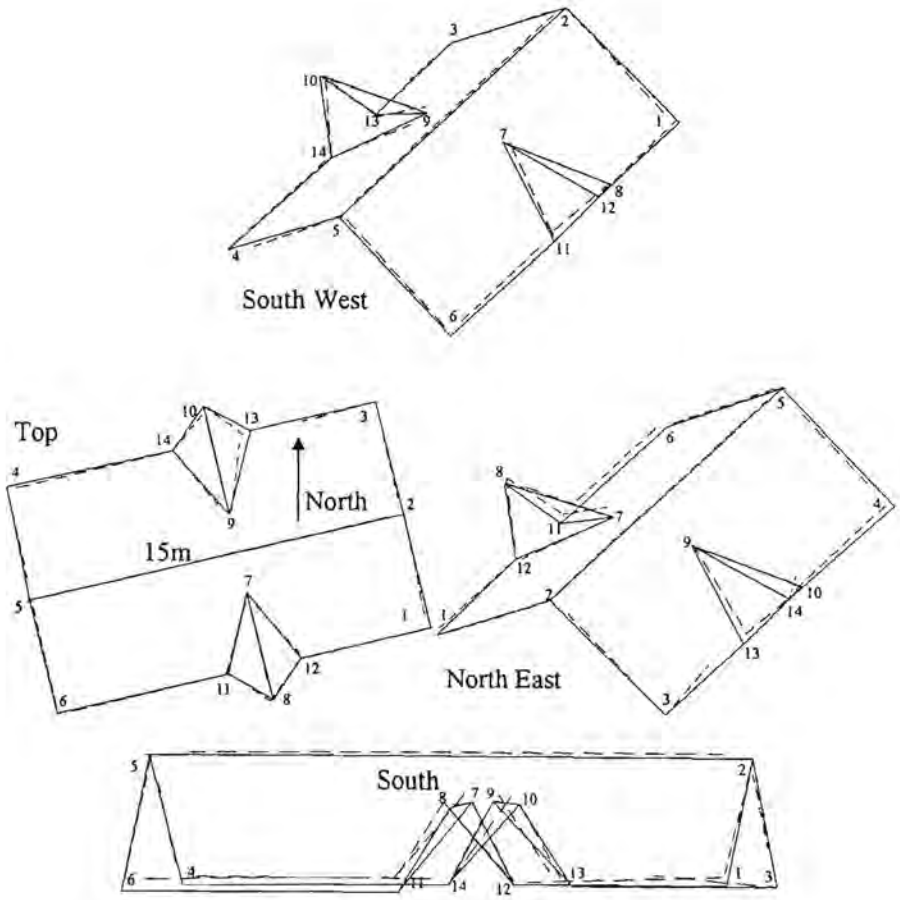


Figure 21: Roof 1 – Ground truth and reconstructed lines

Line	δ_1(m)	δ_2(m)	θ(degrees)
1-2	.15	.06	1.97
2-3	.10	.06	1.32
3-13	.35	.41	1.96
14-4	.14	.23	1.02
4-5	.16	.11	1.86
5-6	.12	.08	.56
6-11	.30	.35	.92
12-1	.05	.19	1.71
2-5	.22	.18	.31
7-8	.01	.08	1.68
9-10	.01	.02	.36
7-11	.38	.32	2.19
7-12	.02	.08	1.94
8-11	.23	.34	3.34
8-12	.07	.05	1.06
9-13	.23	.28	6.8
9-14	.13	.14	.51
10-13	.14	.25	3.17
10-14	.14	.25	3.17

Table 4 : Roof 1 - Comparison between ground truth and reconstructed lines

The actual separation between ground truth and reconstructed lines was also computed. This comparison was carried out in the same manner as the comparison of

the lines obtained from the two algorithms in Section 4.3.3. The only difference is that the line represented by P_1 and P_2 was converted to point and direction vector form using the point P_1 and a direction vector comprising three direction cosines (Equation 60). The distances between the ground truth co-ordinates for the roof corners and the determined lines were then computed.

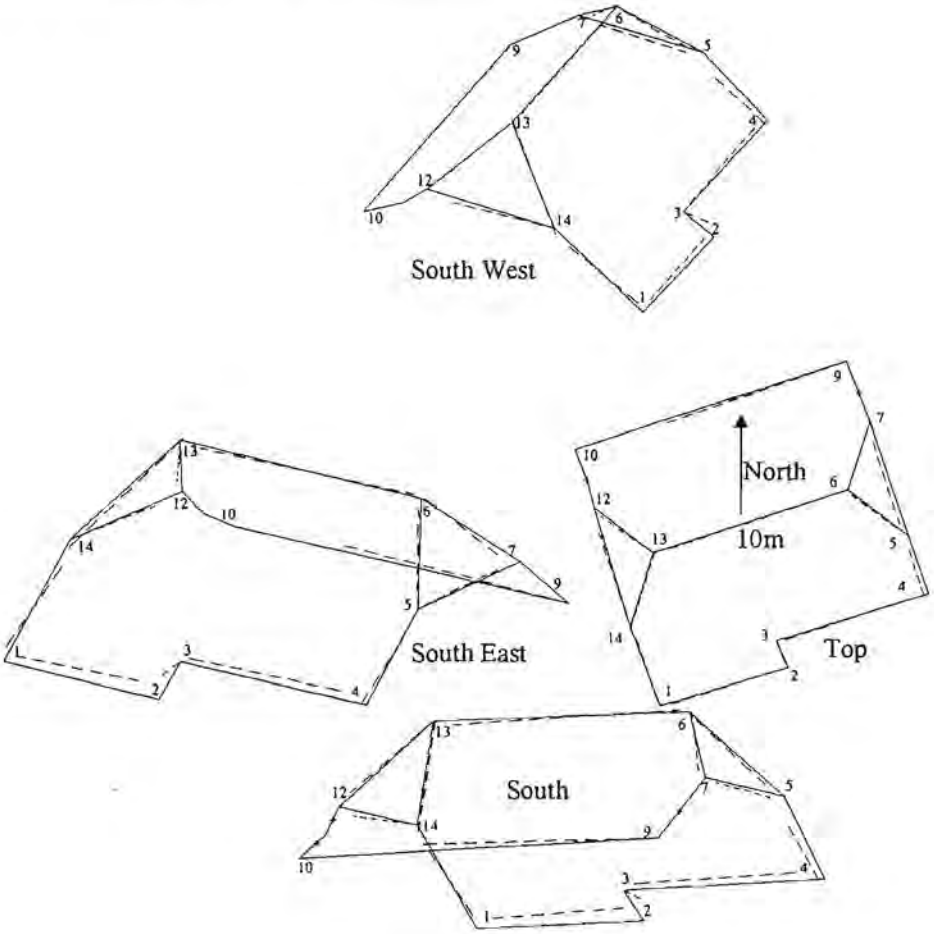


Figure 22: Roof 2 - Ground truth and reconstructed lines

Line	δ_1(m)	δ_2(m)	θ(degrees)
1-2	.34	.47	1.94
2-3	.49	.14	20
3-4	.32	.38	1.25
4-5	.18	.35	6.28
5-6	.13	.08	1.75
6-13	.19	.06	1.36
6-7	.08	.24	5.34
9-10	.29	.32	.21
12-13	.16	.17	2.91
12-14	.21	.15	1.32
14-1	.23	.23	7.29
13-14	.06	.17	1.62

Table 5 : Roof 2 - Comparison between ground truth and reconstructed lines

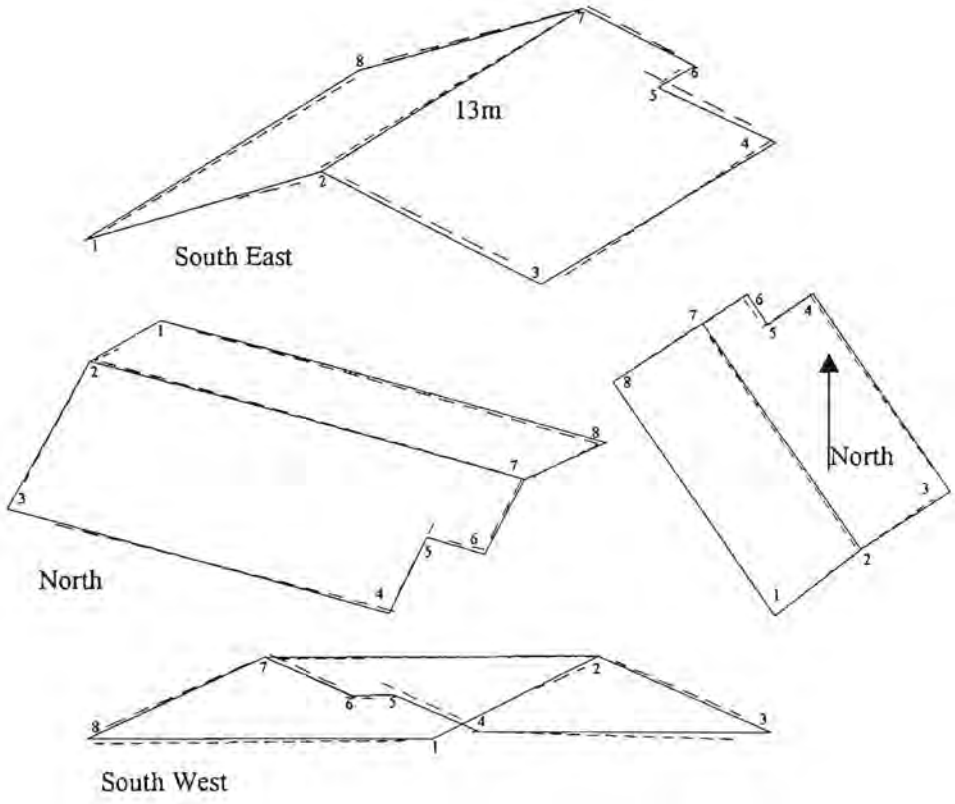


Figure 23: Roof 3 - Ground truth and reconstructed lines

Line	δ_1(m)	δ_2(m)	θ(degrees)
1-2	.10	.11	4.86
2-3	.08	.15	1.12
3-4	.38	.23	14
4-5	.11	.18	1.7
5-6	.24	.26	2.47
6-7	.08	.11	1.05
7-8	.10	.16	1.03
8-1	.11	.05	.30
2-7	.18	.10	.26

Table 6: Roof 3 - Comparison between ground truth and reconstructed lines

Comments and discussion on the above results are given in Chapter 5

4.3.4.2. Close Range Case

Figure 24 shows the reconstructed lines, together with the circular targets (control points) plotted in their co-ordinated positions. The scale of the images was approximately 1:150 with a ground resolution of 1.3mm. Images were captured from a combination of three different height levels, and three horizontal positions.

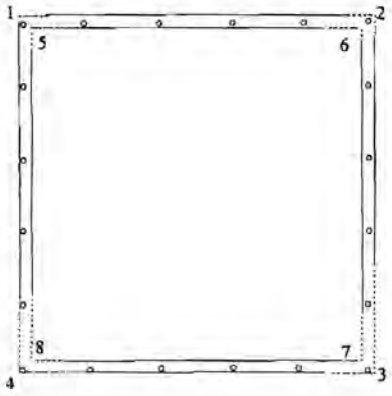


Figure 24: Reconstructed lines on metal frame (circular targets shown in given positions)

Table 7 below provides the standard deviations and χ^2 test results for lines reconstructed in this case. σ_o was given the value 7.0E-6.

Line	σ_X	σ_Y	σ_Z	σ_o	χ^2 test
1-2	0	.604	1.08	.0165	1
2-3	0	.537	.923	.0145	1
3-4	.523	0	.749	.0153	1
6-7	.478	0	.73	.014	1
3-4	0	1.09	1.81	.0195	1
7-8	0	1.35	2.35	.0229	1
4-1	.749	0	1.28	.02	1
8-5	.609	0	1.12	.018	1

Table 7: Standard deviations (mm) and χ^2 test results for reconstructed lines in close-range test

In general the results are consistent for all line orientations, indicating that the camera placement provided suitable geometry for the reconstruction of all lines.

5. Comments and Discussion on findings

The aerial test case deals with a number of lines with different lengths, orientations and lighting characteristics. The discrepancies between reconstructed lines and ground truth therefore, are quoted here in ranges of values as opposed to maximum values and r.m.s., as it is believed that such generalisation would be misleading in these circumstances.

It can be seen in the diagrams and tables in Section 4.3.4 that most measured discrepancies between lines are between 5cm and 35cm. It can also be seen, particularly in the case of Roof 2 that the correspondence between reconstructed lines and the ground truth is better on lines that constitute boundaries between roof panels. See for example the statistics for lines 7-8, and 9-10 in Roof 1, 5-6, 6-7 and 6-13 in Roof 2, and 2-7 in Roof 3. Discrepancies from ground truth lines are generally slightly smaller for these lines (between 5cm and 25cm as opposed to 45cm). These lines are defined by variations in light intensity with uniform texture on either side, giving a clear boundary that is easily observed. More difficulty was experienced in observing roof eves, which were often difficult to distinguish against background shadows, fascia boards and the base of the building wall. Some of these lines were difficult to differentiate from edges of terraces at ground level.

In this case no significant correlation between the length of the lines and the accuracy obtained could be observed. Shorter lines (for example 7-8, 9-10, and 8-11 in Roof 1, 5-6 and 13-14 in Roof 2) of 2 to 3 meters that were well defined in the images and easily observable, could be reconstructed with no less accuracy than the longer lines (e.g. 2-5 in Roof 1, 6-13 in Roof 2, and 2-7 in Roof 3) of 10 to 15 meters. Shorter lines that were not easily distinguishable on the images due to shadows or other reasons did not correlate well with the ground truth data for that reason. Examples include 9-13 and 10-13 in Roof 1, 2-3 in Roof 2, and 4-5 and 5-6 in Roof 3.

5.1. Some Practical Features of the Algorithm

During this study, a few practical features relating to these algorithms became evident.

The POP method has an advantage in that the operation takes place directly in the region of interest, unlike other algorithms that make use of the line-point closest to the

origin. For example, if a rooftop is measured, the two points computed by the POP method can be on the rooftop and near the endpoints. This is an advantage in terms of interpretability and the readiness of the data for use in further application. If endpoints themselves are observed in an image used in the calculation of provisional values, the final values can be used as true endpoints in applications such as GIS where high accuracy is not always required. Furthermore, the standard deviations of the points determined in the POP algorithm are directly relevant and easily interpreted since they provide precision estimates for the X, Y and Z fix of two points on the line in the region of interest. Special treatment is required to obtain standard deviations of a point in the region of interest when Mulawa's method is used, since the standard deviations apply to a point that could be far away from the region of interest. The standard deviations of the direction vector are more useful, but not readily interpreted as the further the line centre point and unit vector are from the region of interest, the greater the exaggeration of errors when extrapolating.

A possible answer to this problem in Mulawa's method, is the use of a point on the line segment in question as opposed to the point closest to the origin. The point could be determined provisionally using the method described in Section 3.2.3. This would involve calculating a projection plane through the perspective centre and two observations in an image, and intersecting a single observation ray from the second image with that plane. Subsequently, either an X, Y or Z value (chosen according to the criteria given in Section 3.2.2 for selecting which co-ordinate to hold fixed) from that provisional point is held fixed to ensure there is only one solution.

From a geometrical interpretation point of view, the discrepancy vector **f** in the POP adjustment provides values for the distances from the points to the projection planes, giving a direct indication of the magnitude of errors in object space.

Since this method using a pencil of planes is based directly on object space geometry, the algorithm offers opportunity for customisation. For example, the concept of using projection planes directly can be adapted for the reconstruction of industrial objects such as pipes and cylinders. In these situations, object lines observed in any particular image do not correspond with the observations from other images. Work in this field is presently being undertaken by Dingle (1997). In this situation, the regularity of

pipes or cylinders can be exploited by the use of a mean projection plane from each image.

An advantage that Mulawa's method holds over the POP method as it has been described, is that a single observation of the line in a particular image can be included in the adjustment. The POP method requires a minimum of two observations in each image in order to generate projection planes. A possible way to overcome this problem is to substitute one of the provisional points ($\mathbf{P}_1$ or $\mathbf{P}_2$ in Figure 15, whichever corresponds least with the single observation), for the second observation. This point should be allocated a weight of zero. Using the observation and the point in object space, a plane can be formed, allowing the single observation to be included in the adjustment.

A further possible field to investigate is the application of this method to theodolite observations. No fictitious image would be required as a plane can be calculated using the telescope centre and two observations (each including azimuth and zenith angles).

6. Conclusions

This thesis has given a background to the theory of line photogrammetry, outlining the fundamental concepts involved, and describing some of the various approaches to the subject. A means of determining the equations of a 3D line in space using observations in two or more oriented images is presented. When three or more images are used, the concept of a best-fit line through a 'pencil of planes' is employed to calculate the positions of two points in space that represent the 3D line.

Tests were done using a set of aerial images with a scale of 1:5000 and a pixel size of 15μm. A method of comparing computed lines and lines on the ground given by ground-truth data, uses the perpendicular distances from the two points representing one line, to the second line, which is represented by a point and a direction vector. Results showed that for lines 2m to 15m in length and clearly defined in the images, computed line equations deviated from the ground truth by between 5cm and 25cm. Most reconstructed lines deviated from the ground truth by less than 35cm with the most poorly defined line deviating by 0.5m, where 0.5m on the ground is represented by 0.1mm on the image. These results confirm that the POP algorithm is a practicable means of adjusting observations to obtain best-fitting 3D lines using observations made in a set of oriented images.

Overall, it is concluded that the presented method of computing 3D line equations has potential as a means of reconstructing rectilinear objects more rapidly than is possible using point-based photogrammetry. This is possible when one considers the observation time required to observe individual points that correspond between images, or to run an image matching routine to identify commonly observed points, is much greater than it would be if just two single points on a line feature are required from each image.

An advantage of the POP method over other methods of line photogrammetry is that the output consists of two points situated directly in the region of interest (i.e. on the line segment itself). This is helpful when the calculated lines are to be consolidated, extended or trimmed in the process of 3D solid reconstruction.

Although the close-range case was tested to a lesser extent than the aerial case, the results were considered sufficient to prove the POP algorithm to be a viable method of reconstructing straight lines at close range.

With regard to the comparison of results obtained by the POP and Mulawa's method, lines reconstructed using the 'pencil-of-planes' method proved to differ by no more than a few millimeters (less than 0.2 pixels) from the lines reconstructed using Mulawa's method.

In general, it has been demonstrated that the concept of line photogrammetry still has the potential to be broadened through the investigation into alternative methods. Although after this work, it cannot be concluded that the POP algorithm is a major improvement on other methods, its existence and viability emphasises the importance of further research into the subject.

7. References

Andreson, K. and Helsch, R. (1990): *Calculation of Analytical Elements in Space Using a Contour Algorithm*, SPIE Vol. 1395 Close Range Photogrammetry Meets Machine Vision, pp. 863-869

Cooper, M.A.R., and Robson, S. (1996): *Theory of Close Range Photogrammetry*, Close Range Photogrammetry and Machine Vision, Department of Photogrammetry and Surveying, University College London, Whittles Publishing, pp.9-51.

Chang, Y.-L., and Aggarwal, J.K. (1990): *Reconstructing 3D lines from a Sequence of 2D Projections: Representation and Estimation*, IEEE, ch. 2934, pp. 101-105.

Deren, -I. L., and Guoqing, Z. (1994): *CAD-Based Line Photogrammetry for Automatic Measurement and Reconstruction of Industrial Objects*, ISPRS Commission V Symposium, Melbourne, Australia, IAP Vol. 30(5), pp. 231-240.

Dingle, M. R. (1997): Phd Thesis in preparation, Department of Geomatics, University of Cape Town.

Dresden, A. (1930): *Solid Analytical Geometry and Determinants*, Swathmore College.

Fraser, Clive S. (1992): *Photogrammetric Camera Component Calibration - A review of Analytical Techniques*, Workshop on Calibration and Orientation of Cameras in Computer Vision (TU-1), XXV11 ISPRS Congress, Washington DC.

Fraser, C., and Dawson, J. (1996): *Object Reconstruction of Circles in Space from Non-Homologous Edgepoints*, Geomatics Research Australasia, No. 64, pp. 33-46.

Fryer, J.G. (1996): *Camera Calibration*, Close Range Photogrammetry and Machine Vision, Department of Photogrammetry and Surveying, University College London, Whittles Publishing, pp.156-179.

Gülch, E. (1995): *Line Photogrammetry- A Tool for Precise Localization of 3D Points and Lines in Automated Object Reconstruction*, SPIE, Vol. 2486, pp.2-12.

Heikkinen, Jussi. (1992): *The use of Linear Features as Reference Datum in Digital Map Revision, Proceedings*, 17th ISPRS Congress, Washington DC, Commission IV.

Kubik, K. (1991): *Relative and Absolute Orientation Based on Linear Features*, ISPRS Journal of Photogrammetry and Remote Sensing, 46, pp. 199-204.

Mason, S.O., and Streilein, A. (1996): *Photogrammetric Reconstruction of Buildings for 3D City Models*, SA J. Surveying & Mapping, 23(5): 244-262.

Mason, S., and Baltsavias, E. (1997): *Image-Based Reconstruction of Informal Settlements, 2nd Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Ascona, Switzerland.

Mason, S. (1996): *3D Building Reconstruction using Composites of Surface Primitives: Concept*, XVIII Congress, Vienna, July.

Mikhail, E.M. (1976): *Observations and Least Squares*, IEP, New York.

Mikhail, E.M. (1993): *Linear Features for Photogrammetric Restitution and Object Completion*, Proceedings SPIE, Vol. 1944, pp. 16-24.

Mulawa, D.C., and Mikhail, E.M. (1988): *Photogrammetric Treatment of Linear Features*, ISPRS 16th Congress, Commission III, Kyoto, Japan.

Nevatia, R. and Babu, R. (1980): *Linear Feature Extraction and Description*, Computer Graphics and Image Processing 13, pg. 257-269.

Petsa, E., and Patias, P. (1994): *Formulation and Assessment of Straight Line Based Algorithms for Digital Photogrammetry*, ISPRS Commission V Symposium, Melbourne, Australia, IAP, Vol. 30(5), pp.310-317.

Petsa, E., and Patias, P. (1994a): *Relative Orientation of Image Triples Using Straight Linear Features*, Proceedings ISPRS Commission III, Munich, pp.663-669.

Patias, P., Petsa, E., and Streilein, A. (1995): *Digital Line Photogrammetry, Institute of Geodesy and Photogrammetry*, ETH Zurich, Report 252.

Schwermann, R. (1994): *Automatic Image Orientation and Object Reconstruction Using Straight Lines in Close Range Photogrammetry*, International Archives of Photogrammetry and Remote Sensing, Vol. 30(5), pp. 349-356.

Smit, J. L. (1997): *Three Dimensional Measurement of Textured Surfaces Using Digital Photogrammetric Techniques*, Phd Thesis, Department of Geomatics, University of Cape Town.

Tommaselli, A. M. G., and Tozzi, C.L. (1996): *A Recursive Approach to Space Resection Using Straight Lines*, PE&RS, Vol. 62, No.1, pp. 57-66.

Van der Heijden, Ferdinand. (1995): *Edge and Line Feature Extraction Based on Covariance Models*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, No. 1.

Wilkin, A. (1992): *Robust 3D-Object Representation by Linear Features*, International Archives of Photogrammetry and Remote Sensing, Vol. 29(5), pp. 659-666.

Zielinsky, H. (1993): *Object Reconstruction with Digital Line Photogrammetry*, Doctoral thesis, Department of Geodesy and Photogrammetry, KTH, Stockholm.

Zhou,Y.T;Venkateswar,V.,and Chellappa, R. (1989): *Edge Detection and Linear Feature Extraction Using a 2D Random Field Model*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 1

8. Appendix

The following tables show the standard deviations (σ) of the X, Y and Z co-ordinates of both points in object space for each reconstructed line. The standard deviation of an observation of unit weight a posteriori (σ_o), and the results of the χ^2 test (by means of a one or zero) at a confidence level of 99% are also given. The final column indicates the number of an image if it was excluded from that particular adjustment.

Roof 1 $\sigma_{o\ a\ priori}$ = 7.5E-6

Line	σ_X	σ_Y	σ_Z	σ_o	χ^2 test	Image Excl.
1-2	.009	0	.02	2.38E-06	0	
2-3	.023	0	.049	6.57E-06	1	
2-3	.021	0	.040	5.12E-06	1	88
4-5	.099	0	.181	1.47E-05	1	97
4-5	.092	.202	0	2.79E-05	1	89
5-6	.005	0	.010	1.15E-06	0	
5-6	.014	0	.024	1.64E-06	0	97
2-5	0	.042	.111	1.64E-05	1	
2-5	0	.027	.063	8.21E-06	1	97
7-8	.005	0	.011	1.20E-06	0	
7-8	.013	0	.024	1.53E-06	0	97
9-10	.008	0	.017	2.12E-06	0	
7-11	.027	0	.105	7.52E-06	1	89
7-12	.029	0	.055	6.31E-06	1	
8-11	.169	.108	0	1.89E-05	1	89
8-12	0	.064	.125	1.07E-05	1	
9-13	.001	0	.004	4.16E-07	0	89
10-13	.289	.146	0	2.78E-05	1	
10-14	.076	.122	0	1.22E-05	1	89
6-11	0	.033	.086	1.31E-05	1	
12-1	0	.040	.111	1.62E-05	1	
3-13	0	.057	.156	1.88E-05	1	
14-4	0	.047	.115	1.36E-05	1	

Roof 2 $\sigma_{o\ a\ priori}$ = 7.5E-6

Line	σ_X	σ_Y	σ_Z	σ_o	χ^2 test	Image Excl.
1-2	0	.017	.039	5.7E-6	1	97
	0	.035	.09	1.4E-5	1	
2-3	.008	0	.017	1.9E-6	0	
3-4	0	.043	.111	1.7E-5	1	
	0	.007	.016	2.2E-6	0	97
4-5	.035	0	.075	8.9E-6	1	
	.04	0	.08	8.1E-6	1	88
5-6	.25	.13	0	2.3E-5	1	
6-13	0	.058	.148	2.3E-5	1	

	0	.013	.028	3.9E-6	1	97
13-14	.014	.048	0	4.9E-6	1	
14-1	.034	0	.072	7.7E-6	1	
	.037	0	.066	7.0E-6	1	88
12-13	0	.055	.067	1.3E-5	1	
	0	.057	.052	8.8E-6	1	88
12-14	.067	0	.198	1.4E-5	1	89nv
6-7	.015	0	.027	4.1E-6	1	
8-11	0	.037	.096	1.5E-5	1	
	0	.001	.002	2.6E-7	0	97
9-10	0	.046	.115	1.7E-5	1	
	0	.002	.004	5.5E-7	0	97
5-8	.082	0	.165	1.9E-5	1	
	.05	0	.159	1.2E-5	1	88

Roof 3 $\sigma_{o\ a\ priori}$ = 7.5E-6

Line	σ_X	σ_Y	σ_Z	σ_o	χ^2 test	Image Excl.
1-2	0	.04	.072	1.7E-5	1	
2-3	0	.061	.172	1.8E-5	1	
3-4	.022	0	.044	3.9E-6	1	97
4-5	0	.04	.092	1.5E-5	1	
5-6	.052	0	.126	1.6E-5	1	
6-7	0	.041	.097	1.4E-5	1	
7-8	0	.023	.036	5.9E-6	1	97
8-1	.036	0	.081	8.8E-6	1	
2-7	.025	0	.057	6.2E-6	1	

The Expanded Coplanarity Condition as Given in Equation 13

$$-N X_1 Y_S + N X_1 Yo + N X_S Y_1 - N Xo Y_1 + M X_1 Z_S - L Y_1 Z_S - M X_1 Zo + L Y_1 Zo - M X_S Z_1 + M Xo Z_1 + L Y_S Z_1 - L Yo Z_1 = 0$$

The Expanded Coplanarity Condition as Given in Equation 40

$$-X_1 Y_0 Z + X_2 Y_0 Z + X_0 Y_1 Z - X_2 Y_1 Z - X_0 Y_2 Z + X_1 Y_2 Z + X_1 Y Z_0 - X_2 Y Z_0 - X Y_1 Z_0 + X_2 Y_1 Z_0 + X Y_2 Z_0 - X_1 Y_2 Z_0 - X_0 Y Z_1 + X_2 Y Z_1 + X Y_0 Z_1 - X_2 Y_0 Z_1 - X Y_2 Z_1 + X_0 Y_2 Z_1 + X_0 Y Z_2 - X_1 Y Z_2 - X Y_0 Z_2 + X_1 Y_0 Z_2 + X Y_1 Z_2 - X_0 Y_1 Z_2 = 0$$